A Maximin Approach to Finding Fair Spanning Trees

Andreas Darmann, Christian Klamler, and Ulrich Pferschy

Abstract

This paper analyzes the computational complexity involved in solving fairness issues on graphs, e.g., in the installation of networks such as water networks or oil pipelines. Based on individual rankings of the edges of a graph, we will show under which conditions solutions, i.e., spanning trees, can be determined efficiently given the goal of maximin voter satisfaction. In particular, we show that computing spanning trees for maximin voter satisfaction under voting rules such as approval voting or the Borda count is $\mathcal{NP}$-hard for a variable number of voters whereas it remains polynomially solvable for a constant number of voters.

1 Introduction

Spanning trees have first been used in connection with fair division problems in the 1970s for fairly assigning costs to individuals in a graph theoretical setting (Bird [3]). From this starting point, a huge body of literature has developed in recent years with a certain vicinity to Social Choice Theory, often axiomatically motivated (e.g., Bogomolnaia and Moulin [4], Dutta and Kar [11] and Kar [12]). In this paper we want to strengthen this link to Social Choice Theory by looking at the maximin voter satisfaction and analyzing the computational complexity of solution methods based on certain well-known social choice rules.

Many of the current papers use graphs to model certain networks, such as the installation of water or power networks, oil pipelines, road constructions, or links between different countries. Costs are assigned to the edges in such a graph and the goal is to connect all nodes (individuals, countries, etc.) at minimum total cost and fairly assign that cost to the nodes.

In this paper[1] we do not consider any monetary costs, be it because they are negligible or because they are covered by some external source (e.g., the state). Our approach is based on individuals' preferences over the edges of a graph and we analyze methods that - given those preferences - fairly, i.e., socially acceptably, install networks. The focus of our analysis, however, does not lie in the quality of the solution, i.e., in an axiomatic analysis of the solution methods, but in the computational complexity involved.

An example in that respect could be a village that has to install a sewage or water network or countries that need to agree on oil pipelines. Each homeowner or country needs to be connected but obviously there are many different ways to connect everyone. Mathematically the situation can be represented as a graph, i.e., the nodes are the homeowners and the edges are the connections between pairs of homeowners, and a solution is a spanning tree. The problem, however, is that homeowners might have different preferences over which connections (edges) should be used in the spanning tree. E.g. one homeowner might prefer a certain connection over another connection for environmental reasons, whereas another homeowner might just prefer any connection further away from his own garden to any connection that is closer to his garden. As we consider that costs are no issues here, the ordinal rankings over edges by those homeowners are the only inputs that can be used by any solution method.

[1] A major part of this work appeared in Darmann et al. [9, 10].

The quality of different solution methods based on social choice rules has been analyzed in a previous paper by Darmann et al. [8], extensive studies of social choice rules can be found in Brams and Fishburn [6], Nurmi [14] and Saari [19] among many others. The goal in this paper, however, is to look at the computational complexity involved in finding optimal spanning trees based on such solution methods, i.e., whether such solutions can be found in polynomial time or not.[2] Our main focus will be on methods using scores as in the Borda count or in approval voting and the basis for evaluating different solutions will be the maximin voter satisfaction (MMVS). In a completely different setup, namely the consideration of different scenarios to represent uncertainty in Robust Optimization, closely related models of spanning tree problems were considered, e.g., in Aissi et al. [1] and Kouvelis and Yu [13]. While their works assign arbitrary numerical values as weights of the edges, we will consider the outcome of voting procedures to compare edges and trees.

An important differentiation arises from the number of voters considered in the problem, i.e., whether this number is fixed or not. Following the results of Aissi et al. [1], it is shown that for a fixed number of voters, solutions based on MMVS can be found in polynomial time. Things do change when the number of voters is variable, i.e., the number of voters is part of the input of the problem. This makes the problem significantly harder in the case of general edge weights as has been shown by Kouvelis and Yu [13]. However, as far as the $\mathcal{NP}$-hardness results are concerned, the simple structure of edge weights arising from the respective voting rules requires a completely different proof technique than their previously known results.

The contribution of this paper is to answer the questions of complexity posed by the application of voting rules from Social Choice Theory. We show that even under very simple voting structures such as approval voting, vote-against-t elections and choose-t elections for $t \geq 2$, MMVS is $\mathcal{NP}$-hard. Furthermore we show that MMVS is intractable for both dichotomous and multichotomous voter preferences. Moreover, irrespective of whether the voters' preferences are weak or strict orders on the edge set, MMVS under Borda voting is $\mathcal{NP}$-hard. Only for the two structurally most simple solution methods under consideration MMVS can be solved in polynomial time, namely for plurality voting and vote-against-1 election. In fact, our result settles the complexity status for any reasonable election process: If every voter is allowed to distinguish only one edge in a positive or negative sense the problem remains polynomially solvable. As soon as two or more edges receive an appraisal different from the remaining edges, the problem becomes $\mathcal{NP}$-hard.

The paper is structured as follows: We give the formal framework in Section 2 and then restate and discuss previous results for a fixed number of voters in Section 4. In Section 5 we keep the number of voters variable and prove our main results.

2 Preliminaries

In order to be able to express preferences, we give some basic definitions for relations; the terminology is adopted from Roberts [17].

A binary relation $\succsim \subseteq A \times A$ on a set A is called *complete* if $\forall a, b \in A,\ a \neq b,\ (a \succsim b \text{ or } b \succsim a)$. $\succsim$ is *reflexive* if $\forall a \in A,\ a \succsim a$. It is called *transitive* if $\forall a, b, c \in A,\ (a \succsim b \text{ and } b \succsim c) \Rightarrow a \succsim c$. Finally, $\succsim$ is called *asymmetric* if $\forall a, b \in A,\ a \succsim b \Rightarrow \neg(b \succsim a)$; and we call it *symmetric* if $\forall a, b \in A,\ a \succsim b \Rightarrow b \succsim a$. A relation is called *weak order* if it is complete, reflexive and transitive. A relation is called *strict order*, if it is complete, transitive and asymmetric.

Let $G = (V, E)$ be an undirected and connected graph. Let $n := |V|$ and τ be the set of spanning trees of G. For every voter i, $1 \leq i \leq k$, we are given a preference relation $\succsim_i$ on E. Unless otherwise stated, $\succsim_i$ is assumed to be a weak order on E, consisting of an asymmetric part $\succ_i$ and

[2] $\mathcal{P} \neq \mathcal{NP}$ is tacitly assumed throughout this paper.

a symmetric part $\sim_i$ respectively. The symmetric part $\sim_i$ of $\succsim_i$ induces a partition $E_1, E_2, ..., E_q$ of E, such that for all j, $1 \le j \le q$, we have $e \sim_i f$ for all $e, f \in E_j$. The sets E_j, $1 \le j \le q$, are called preference classes. In case $q = 2$ we call $\succsim_i$ dichotomous. If $q \ge 3$ the order $\succsim_i$ is called multichotomous. Furthermore, we refer to the k-tuple $\pi = (\succsim_1, \succsim_2, \ldots, \succsim_k)$ as a voter preference profile.
The basic concept used in this work is the one of voters' scoring functions, which can be understood as a generalization of the positional scoring procedures (for details concerning these procedures see Brams and Fishburn [6]).

Definition 2.1 *Let $1 \le i \le k$. We call a function $v_i : E \to \mathbb{N}_0$ voter i's scoring function, if*

1. *for all $e, f \in E$ $e \succsim_i f \;\Leftrightarrow\; v_i(e) \ge v_i(f)$, and*
2. *$\max_{e \in E}\{v_i(e)\}$ is bounded by a polynomial in n.*

Definition 2.2 *For $1 \le i \le k$ let v_i be voter i's scoring function. Voter i's score (or count) of tree $T \in \tau$ is $v_i(T) := \sum_{e \in T} v_i(e)$.*

Hence, voters' preferences on trees are assumed to be additively separable, i.e., there do not exist complementaries or synergies between the edges. Many scoring procedures can be embedded in the framework of voters' scoring functions. For example, approval voting (see Brams and Fishburn [5]), plurality voting (see Roberts [18]), vote-against-t elections (presented in Brams and Fishburn [6]) and Borda voting (see Brams and Fishburn [6] and Vorsatz [22]) can be formulated within this framework.[3]

Definition 2.3 *Let $1 \le i \le k$. For $e, f \in E$, $e \ne f$, let*

$$\delta_i(e,f) := \begin{cases} 2 & \text{if } e \succ_i f \\ 1 & \text{if } e \sim_i f \\ 0 & \text{otherwise.} \end{cases}$$

Then in Borda voting, voter i's scoring function is the Borda function $b_i : E \to \mathbb{N}_0$ defined by $b_i(e) := \sum_{f \in E \setminus \{e\}} \delta_i(e,f)$. For $e \in E$ we call $b_i(e)$ voter i's Borda[4] count of edge e. Voter i's Borda count of tree $T \in \tau$ is $b_i(T) := \sum_{e \in T} b_i(e)$.

In approval voting, for every voter i the set E is partitioned into a set $S_i \subseteq E$ of edges voter i approves of and a set $S_i^c := E \setminus S_i$ of edges voter i disapproves of.

Definition 2.4 *Let $1 \le i \le k$. In approval voting voter i's scoring function is the function $a_i : E \to \mathbb{N}_0$ with*

$$a_i(e) = \begin{cases} 1 & \text{if } e \in S_i \\ 0 & \text{if } e \in S_i^c. \end{cases}$$

[3]The use of scoring functions on edges to obtain scores for spanning trees has not received much attention yet in the literature. A general axiomatic analysis as surveyed by Barbera et al. [2] might help to provide support for such a use.

[4]If $\succ_i$ is a strict order on E, we have $b_i(e) = 2 \cdot |\{f \in E : e \succ_i f\}|$ for $e \in E$. Let $\hat{\delta}_i(e,f) := \frac{1}{2}\delta_i(e,f)$ for all $e, f \in E$, $e \ne f$, and let $\hat{b}_i(e) := \sum_{f \in E \setminus \{e\}} \hat{\delta}_i(e,f)$ for $e \in E$. Thus $\hat{b}_i(e) = |\{f \in E : e \succ_i f\}|$, and hence $\hat{b}_i(e)$ would define voter i's Borda count of edge e in the canonical way. Note that $b_i(e) > b_i(f) \Longleftrightarrow \hat{b}_i(e) > \hat{b}_i(f)$ for all $e, f \in E$, $e \ne f$, and $\sum_{e \in T_1} b_i(e) > \sum_{e \in T_2} b_i(e) \Longleftrightarrow \sum_{e \in T_1} \hat{b}_i(e) > \sum_{e \in T_2} \hat{b}_i(e)$ for all $T_1, T_2 \in \tau$. The function $\hat{b}_i$ however does not map from E into the set of non-negative integers but may take rational values as well. Since this causes some technical inconvenience (i.e., Theorem 4.1 cannot be applied directly), $\hat{b}_i$ is omitted in this work.

The function a_i is called voter i's approval function. Voter i's approval count of $T \in \tau$ is defined by $a_i(T) := \sum_{e \in T} a_i(e)$.

Choose-t elections and vote-against-t elections constitute two special cases of approval voting. A *choose-t election*[5] corresponds to approval voting subject to the requirement that for a fixed $t \in \mathbb{N}$ $|S_i| = t$ for $1 \leq i \leq k$. In this context, a choose-1 election is called *plurality voting.* Approval voting under the requirement that for a fixed $t \in \mathbb{N}$ $|S_i^c| = t$ for $1 \leq i \leq k$ is called *vote-against-t election.*

3 Problem formulation

With the above preliminaries we are now able to state the maximin voter satisfaction problem.

Definition 3.1 ***Maximin voter satisfaction problem (MMVS)***
Let $G = (V, E)$ be an undirected graph, let I be a set of voters and let π be a voter preference profile. For $i \in I$ let v_i be voter i's scoring function. The maximin voter satisfaction problem (MMVS) is the following problem:

$$\max_{T \in \tau} \min_{i \in I} v_i(T)$$

Maximizing the minimum of such concepts as utility, costs, time, etc. is a very common way to formalize the idea of fairness. Such a maximin approach to fairness can especially be found in the literature on networks, scheduling, etc. On the other hand, maximin fairness also has a certain link to fairness in Social Choice Theory, originally discussed decades ago by Rawls [16]. However, there are also many other approaches to formalize fairness based on proportionality, equitability, envy-freeness, etc. and used in areas such as mathematics and economics (Brams and Taylor [7], Thomson [21]).
From a completely different point of view the problem appears in the Operations Research literature in the context of Robust Optimization. One possibility to model an optimization problem under uncertainty is the consideration of different scenarios each of which induces different data for the problem. Maximizing the objective function for the worst-case scenario amounts to a maximin problem with voters corresponding to scenarios. In this context Aissi et al. [1] refer to an analogon of MMVS as *max-min spanning tree problem* while Kouvelis and Yu [13] use the terminology *absolute robust minimum spanning tree problem.* In this paper, however, the aim is to analyze the complexity of aggregating voters' opinions with the help of special types of voting procedures.

4 MMVS with a fixed number of voters

In this section the number k of voters is assumed to be a constant integer number. Likewise one could say that k is not regarded as a part of the input within this section. With this point of view MMVS is known to be solvable in polynomial time (see Aissi et al. [1]). We restate this result in the following theorem.

Theorem 4.1 *(Aissi et al. [1])*
MMVS can be solved in $\mathcal{O}(n^4 W^k \log W)$ time, where $W \in \mathbb{N}$ is an upper bound for the objective function value.

[5] In the literature, choose-t elections are also called t-approval voting (Peters et al. [15]) or vote-for-exactly-t procedures (Brams and Fishburn [6]).

Noting that for approval voting there is $W \leq n$ and for Borda voting $W \leq 2nm$, this theorem yields the following corollary.

Corollary 4.2 *MMVS under approval voting can be solved in $\mathcal{O}(n^{4+k}\log n)$ time. MMVS under Borda voting can be solved in $\mathcal{O}(n^{4+k}m^k\log n)$ time.*

However, for the special case of plurality voting MMVS can even be solved in linear time.

Proposition 4.3 *MMVS under plurality voting can be solved in $\mathcal{O}(mk) = \mathcal{O}(m)$ time.*

Proof. Given the graph $G = (V, E)$, let $E_1 := \{e \in E | v_i(e) = 1$ for at least one i, $1 \leq i \leq k\}$. If the subgraph $H = (V, E_1)$ is acyclic, then there obviously exists a spanning tree T of G such that $E_1 \subseteq T$ holds. In this case trivially $\max_{T\in\tau}\min_{i\in I} v_i(T) = 1$. If on the other hand H contains a cycle, then clearly there cannot exist a spanning tree T of G with $E_1 \subseteq T$. Thus for each spanning tree T of G there is an edge of E_1 that is not contained in T. Hence for each $T \in \tau$ we have $\min_{i\in I} v_i(T) = 0$ which yields $\max_{T\in\tau}\min_{i\in I} v_i(T) = 0$.
Calculating the set E_1 takes $\mathcal{O}(mk) = \mathcal{O}(m)$ time, the determination whether H is acyclic or not can be done in $\mathcal{O}(m)$ time. This proves the proposition. $\square$

5 MMVS with a variable number of voters

In this section the number k of voters is not assumed to be constant but may vary instead, i.e., k is considered to be part of the input. This approach seems to make MMVS significantly harder. To be more precise, MMVS was shown to be strongly $\mathcal{NP}$-hard for arbitrary scoring functions by Kouvelis and Yu [13]. The question of the computational complexity of MMVS under the common voting rules such as approval voting, plurality voting, choose-t elections, vote-against-t elections and Borda voting is not answered by Kouvelis and Yu [13] though and to the authors' best knowledge has been open so far.
We improve upon the result of Kouvelis and Yu [13] and show that MMVS is $\mathcal{NP}$-hard even in case of very basic voting procedures. In particular, MMVS turns out to be $\mathcal{NP}$-hard even under the simple procedure of approval voting – that is, MMVS remains $\mathcal{NP}$-hard if the range of the voters' scoring functions is restricted to $\{0, 1\}$.[6] We also show that this result still holds if the number of approved or disapproved edges is some fixed $t \geq 2$ (choose-t elections and vote-against-t elections respectively for $t \geq 2$). Moreover, we can show that MMVS is $\mathcal{NP}$-hard under Borda voting. In contrast to these results, it can easily be shown that MMVS under plurality voting and vote-against-1 elections can be solved in polynomial time.
The key instrument used in the $\mathcal{NP}$-hardness proofs presented in this section is to reduce the $\mathcal{NP}$-complete monotone one-in-three 3SAT problem (Schaefer [20]) to the decision problem corresponding to MMVS.

Definition 5.1 ***Monotone one-in-three 3SAT problem (monotone 1-in-3SAT)***

GIVEN:	*A set X of variables and a collection C of clauses over X such that every clause is made up of exactly three positive literals.*
QUESTION:	*Is there a truth assignment for X such that every clause contains exactly one true literal?*

Remark. Note that in above definition every clause contains exactly three literals all of which must be positive. That is, in monotone 1-in-3SAT there are no negated literals. Therefore in monotone 1-in-3SAT the set X of variables corresponds to set of literals over X.

[6]Note that this implies and sharpens the strong $\mathcal{NP}$-hardness result of Kouvelis and Yu [13].

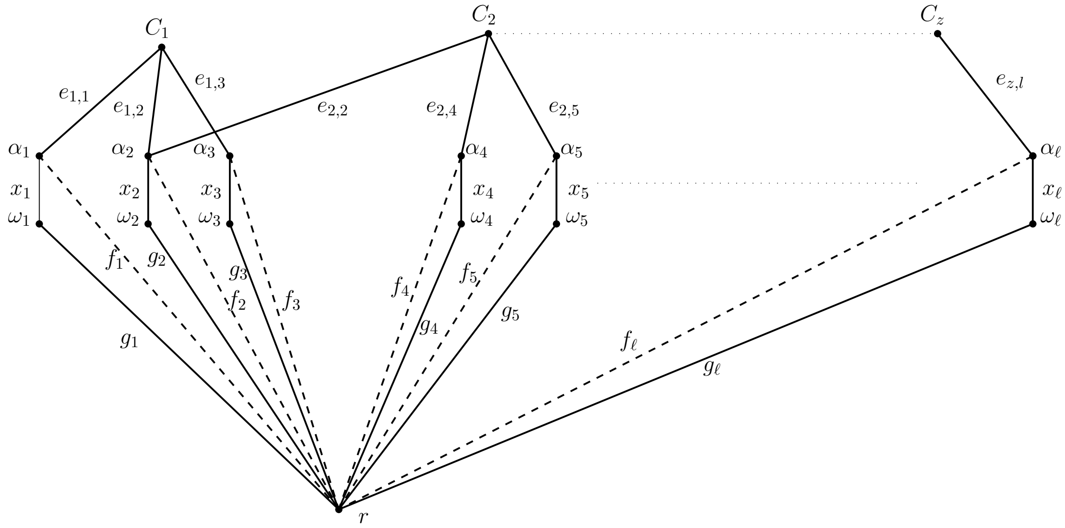

Figure 1: Undirected Graph G derived from instance U_1 of monotone 1-in-3SAT.

5.1 Approval voting and Borda voting

Our first result shows that MMVS is $\mathcal{NP}$-hard already for weak orders if the voters' scoring functions have the simple structure of approval functions.

Theorem 5.1 *Under approval voting MMVS is $\mathcal{NP}$-hard.*

Proof. We will polynomially transform an arbitrary instance of monotone 1-in-3SAT to an instance of MMVS with approval voting.
Let U_1 be an instance of monotone 1-in-3SAT with $X := \{\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_\ell\}$ being the set of variables (= literals) and $C := \{\tilde{C}_1, \tilde{C}_2, \ldots, \tilde{C}_z\}$ being a collection of clauses over X. W.l.o.g. we assume clause $\tilde{C}_1$ to contain the literals $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$. We construct the undirected graph $G = (V, E)$ by the following procedure (see Fig. 1):
Let $V = \emptyset$ and $E = \emptyset$. For each literal $\tilde{x}_j \in X$ add two nodes α_j and ω_j to V. For each clause $\tilde{C}_i \in C$ add node C_i to V. Add node r to V. Next for each literal $\tilde{x}_j \in X$

- add edge x_j to E connecting the nodes α_j and ω_j
- add edge f_j to E connecting α_j and r
- add edge g_j to E connecting ω_j and r
- if $\tilde{x}_j$ is contained in clause $\tilde{C}_i \in C$ add edge $e_{i,j}$ to E connecting the nodes C_i and α_j.

Note that $n = |V| = z + 2\ell + 1$ and $m = |E| = 3\ell + 3z$.
We now establish the voter preference profile π and the corresponding values of the voters' approval functions (see Table 1 and 2). First, we introduce voters χ_j, $1 \leq j \leq \ell$, whose approval functions are given by

$$a_{\chi_j}(e) = \begin{cases} 0 & \text{if } e \in \{x_j, f_j\} \\ 1 & \text{otherwise.} \end{cases}$$

χ_1		χ_2		χ_3		$\cdots$	χ_ℓ	
edge	a_{χ_1}	edge	a_{χ_2}	edge	a_{χ_3}		edge	a_{χ_ℓ}
x_1	$\mathbf{0}$	x_2	$\mathbf{0}$	x_3	$\mathbf{0}$		x_ℓ	$\mathbf{0}$
f_1	$\mathbf{0}$	f_2	$\mathbf{0}$	f_3	$\mathbf{0}$		f_ℓ	$\mathbf{0}$
$e_{1,1}$	$\mathbf{1}$	$e_{1,1}$	$\mathbf{1}$	$e_{1,1}$	$\mathbf{1}$		$e_{1,1}$	$\mathbf{1}$
$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$		$\vdots$	$\mathbf{1}$
$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$		$\vdots$	$\mathbf{1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$		x_{l-1}	$\mathbf{1}$

Table 1: Preference profile of voters χ_j, $1 \leq j \leq \ell$, and the values of the corresponding approval functions.

c_i		$c_i^{j_1}$		$c_i^{j_2}$		$c_i^{j_3}$		$c_i^{f_{j_1}}$		$c_i^{f_{j_2}}$		$c_i^{f_{j_3}}$	
edge	a_{c_i}	edge	$a_{c_i^{j_1}}$	edge	$a_{c_i^{j_2}}$	edge	$a_{c_i^{j_3}}$	edge	$a_{c_i^{f_{j_1}}}$	edge	$a_{c_i^{f_{j_2}}}$	edge	$a_{c_i^{f_{j_3}}}$
x_{j_1}	$\mathbf{0}$	x_{j_1}	$\mathbf{0}$	x_{j_2}	$\mathbf{0}$	x_{j_3}	$\mathbf{0}$	f_{j_1}	$\mathbf{0}$	f_{j_2}	$\mathbf{0}$	f_{j_3}	$\mathbf{0}$
x_{j_2}	$\mathbf{0}$	e_{i,j_2}	$\mathbf{0}$	e_{i,j_1}	$\mathbf{0}$	e_{i,j_1}	$\mathbf{0}$	e_{i,j_1}	$\mathbf{0}$	e_{i,j_2}	$\mathbf{0}$	e_{i,j_3}	$\mathbf{0}$
x_{j_3}	$\mathbf{0}$	e_{i,j_3}	$\mathbf{0}$	e_{i,j_3}	$\mathbf{0}$	e_{i,j_2}	$\mathbf{0}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$
	$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$
other	$\mathbf{1}$	other	$\mathbf{1}$	other	$\mathbf{1}$	other	$\mathbf{1}$	other	$\mathbf{1}$	other	$\mathbf{1}$	other	$\mathbf{1}$
edges	$\vdots$	edges	$\vdots$	edges	$\vdots$	edges	$\vdots$	edges	$\vdots$	edges	$\vdots$	edges	$\vdots$
	$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$		$\mathbf{1}$

Table 2: Preference profile (and corresponding approval functions) derived from clause $\tilde{C}_i$ containing the literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$.

c_1		c_1^1		c_1^2		c_1^3		$c_1^{f_1}$		$c_1^{f_2}$		$c_1^{f_3}$	
edge	a_{c_i}	edge	$a_{c_i^{j_1}}$	edge	$a_{c_i^{j_2}}$	edge	$a_{c_i^{j_3}}$	edge	$a_{c_i^{f_{j_1}}}$	edge	$a_{c_i^{f_{j_2}}}$	edge	$a_{c_i^{f_{j_3}}}$
x_1	$\mathbf{0}$	x_1	$\mathbf{0}$	x_2	$\mathbf{0}$	x_3	$\mathbf{0}$	f_1	$\mathbf{0}$	f_2	$\mathbf{0}$	f_3	$\mathbf{0}$
x_2	$\mathbf{0}$	$e_{1,2}$	$\mathbf{0}$	$e_{1,1}$	$\mathbf{0}$	$e_{1,1}$	$\mathbf{0}$	$e_{1,1}$	$\mathbf{0}$	$e_{1,2}$	$\mathbf{0}$	$e_{1,3}$	$\mathbf{0}$
x_3	$\mathbf{0}$	$e_{1,3}$	$\mathbf{0}$	$e_{1,3}$	$\mathbf{0}$	$e_{1,2}$	$\mathbf{0}$	$e_{1,2}$	$\mathbf{1}$	$e_{1,1}$	$\mathbf{1}$	$e_{1,1}$	$\mathbf{1}$
$e_{1,1}$	$\mathbf{1}$	$e_{1,1}$	$\mathbf{1}$	$e_{1,2}$	$\mathbf{1}$	$e_{1,3}$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$
$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$
$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$
$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$	$\vdots$	$\mathbf{1}$
x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$	x_ℓ	$\mathbf{1}$

Table 3: Preference profile derived from clause $\tilde{C}_1$ which is made up of the literals $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$.

The rest of the voter preference profile is established as follows. Let a clause $\tilde{C}_i \in C$ contain the literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$ – which means node α_y and node C_i are adjacent, $y \in \{j_1, j_2, j_3\}$. Add seven voters denoted by c_i, $c_i^{j_1}$, $c_i^{j_2}$, $c_i^{j_3}$, $c_i^{f_{j_1}}$, $c_i^{f_{j_2}}$ and $c_i^{f_{j_3}}$ to π. Voter c_i assigns value 0 to the edges $x_{j_1}, x_{j_2}, x_{j_3}$ and value 1 to all other edges. Voter c_i^y, $y \in \{j_1, j_2, j_3\}$ assigns value 1 to all edges but to x_y and to the edges $e_{i,u}$ with $u \in \{j_1, j_2, j_3\}$, $u \neq y$, which get value 0. And voter $c_i^{f_y}$, $y \in \{j_1, j_2, j_3\}$, assigns value 0 to the edges f_y and $e_{i,y}$, and assigns value 1 to all the other edges (see Table 2). To illustrate the voter preference profile π, an example is given in Table 3 with the preferences and approval functions of the seven voters corresponding to clause $\tilde{C}_1$ which is made up of the literals $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$.
Having treated all clauses in the way just described the voter preference profile is made up of $k := \ell + 7z$ voters. Note that the instance of MMVS under approval voting defined by $G = (V, E)$, π and the corresponding approval functions can be constructed in polynomial time (with respect to the size of U_1).

Claim 1. There exists a truth assignment for X such that each clause in C contains exactly one true literal if and only if there exists a $T \in \tau$ such that for all p, $1 \leq p \leq k$, $a_p(T) \geq n-2$ holds.

Proof of Claim 1.
"$\Rightarrow$": For a satisfying truth assignment t_S let S be the set of literals set "TRUE" under t_S. Create tree T as follows. Set $T = \emptyset$. For all $\tilde{x}_j \in S$:

- add x_j and g_j to T
- add $e_{i,j}$ to T for all i, $1 \leq i \leq z$, for which edge $e_{i,j} \in G$

For all $\tilde{x}_j \in X \setminus S$, i.e., literals set "FALSE" in t_S, add f_j and g_j to T. Summarizing, we get for $1 \leq j \leq \ell$ the following four properties:

1. $g_j \in T$
2. $x_j \in T \Leftrightarrow \tilde{x}_j$ is set "TRUE" under t_S
3. $x_j \in T \Leftrightarrow e_{i,j} \in T$ for all $i : e_{i,j} \in G$
4. $x_j \in T \Leftrightarrow f_j \notin T$

Since t_S constitutes a satisfying truth assignment, each node C_i, $1 \leq i \leq z$, is connected to node r in T. Obviously, all other nodes of V are connected to r in T as well and thus T is connected. Because of $|T| = |S| + z + \ell + (\ell - |S|) = z + 2\ell$ we get $|T| = n - 1$ and hence the subgraph T is a tree. Due to $|T| = n - 1$ and property 4. we get $a_{\chi_j}(T) = n - 2$ for all $j \in \{1, 2, \ldots, \ell\}$.
As above, let clause $\tilde{C}_i$ be made up of the literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$. The fact that exactly one of the literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$ is set "TRUE" under t_S means exactly one of the edges $x_{j_1}, x_{j_2}, x_{j_3}$ is contained in T. Together with $|T| = n - 1$ this yields $a_{c_i}(T) = n - 2$. Let us now consider the voters $c_i^{j_1}$, $c_i^{j_2}$, $c_i^{j_3}$: W.l.o.g. we may assume that $\tilde{x}_{j_1}$ is set "TRUE" under t_S. Thus $x_{j_1} \in T$, $x_{j_2} \notin T$, $x_{j_3} \notin T$. Due to property 3. we hence get $e_{i,j_1} \in T$, $e_{i,j_2} \notin T$, $e_{i,j_3} \notin T$. This implies

$$a_{c_i^{j_y}}(T) = n - 2$$

for all $y \in \{j_1, j_2, j_3\}$.[7] Finally, properties 3. and 4. yield $a_{c_i^{f_y}}(T) = n - 2$ for all $y \in \{j_1, j_2, j_3\}$.

[7] Clearly, assuming that instead of $\tilde{x}_{j_1}$ either $\tilde{x}_{j_2}$ or $\tilde{x}_{j_3}$ is set "TRUE" under t_S yields $a_{c_i^{j_y}}(T) = n - 2$ as well.

"$\Leftarrow$": Let now Q be a spanning tree with $a_p(Q) \geq n-2$ for all p, $1 \leq p \leq k$. Thus for each voter p in our voter preference profile at most one edge e with $a_p(e) = 0$ is contained in Q. Hence because of voters χ_j the edges x_j and f_j cannot both be contained in Q, $1 \leq j \leq \ell$. Analogously due to voters c_i, $1 \leq i \leq z$, for any clause $\tilde{C}_i$ made up of some literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$ at most one of the edges x_{j_1}, x_{j_2}, x_{j_3} is contained in Q. Next we show that for $1 \leq j \leq \ell$

$$x_j \in Q \Leftrightarrow e_{i,j} \in Q$$

holds for all i with $e_{i,j} \in G$.
Assume $x_j = x_{j_1} \in Q$ and let node C_i be adjacent to nodes $\alpha_{x_{j_1}}$, $\alpha_{x_{j_2}}$ and $\alpha_{x_{j_3}}$ (i.e., in our monotone 1-in-3SAT instance clause $\tilde{C}_i$ is again made up of the literals $\tilde{x}_{j_1}, \tilde{x}_{j_2}, \tilde{x}_{j_3}$). Because of voter $c_i^{j_1}$ we have $e_{i,j_2} \notin Q$ and $e_{i,j_3} \notin Q$. Note that the degree of node C_i equals three and thus $e_{i,j_1} \in Q$ since otherwise C_i would be isolated. Thus $x_j \in Q$ implies $e_{i,j} \in Q$ for all i such that $e_{i,j} \in G$.
On the other hand, let $e_{i,j_1} \in Q$ for some i, $1 \leq i \leq z$, and some j_1, $1 \leq j_1 \leq \ell$. Now $a_{c_i^{j_2}}(Q) \geq n-2$ implies $e_{i,j_3} \notin Q$ and $a_{c_i^{j_3}}(Q) \geq n-2$ implies $e_{i,j_2} \notin Q$. In other words, node C_i is a leaf. Due to voter $c_i^{f_{j_1}}$ we have $f_{j_1} \notin Q$. If there is no u, $1 \leq u \leq z$, $u \neq i$, such that $e_{u,j_1} \in Q$ then it is easy to see that x_{j_1} must be contained in Q since otherwise nodes r and C_i would not be connected. If such an edge e_{u,j_1} is contained in Q, then as a consequence of

$$a_{c_u^{j_1}}(Q) \geq n-2$$

node C_u must be a leaf as well and thus the same argument applies. Hence $x_j \in Q \Leftrightarrow e_{i,j} \in Q$ holds for all i with $e_{i,j} \in G$, $1 \leq j \leq \ell$.
But since node C_i is a leaf, $1 \leq i \leq z$, for each such node there is exactly one j, $1 \leq j \leq \ell$, such that both x_j and $e_{i,j}$ are contained in Q. In other words, the truth assignment $t_{\tilde{S}}$ defined by letting $\tilde{S} := \{\tilde{x}_j | x_j \in Q\}$ be the whole set of literals set "TRUE" under $t_{\tilde{S}}$ is a satisfying truth assignment for the considered instance of monotone 1-in-3SAT. This proves the claim. $\Diamond$
Claim 1 implies that any arbitrary instance of monotone 1-in-3SAT can be reduced to an instance of MMVS under approval voting. As stated before, the instance of MMVS under approval voting can be constructed in polynomial time. Thus it is proven that monotone 1-in-3SAT polynomially transforms to MMVS under approval voting. $\square$

Remark. Note that in case of dichotomous preferences the sets of optimal solutions of MMVS under approval voting and of MMVS under Borda voting obviously coincide.[8] Thus from Theorem 5.1 it follows that, given weak preference orders, MMVS under Borda voting is $\mathcal{NP}$-hard as well.[9]

Proposition 5.2 *MMVS under Borda voting is $\mathcal{NP}$-hard.*

Since dichotomous preferences over the edges induce approval functions in a natural way, it follows from Theorem 5.1 that MMVS is $\mathcal{NP}$-hard for any dichotomous preferences already. Furthermore it can easily be shown that MMVS is $\mathcal{NP}$-hard in the cases of multichotomous preferences as well.

Corollary 5.3 *Let $\pi = (\succsim_1, \succsim_2, \ldots, \succsim_k)$ be a voter preference profile such that $\succsim_i$ is multichotomous for all $1 \leq i \leq k$. Then MMVS is $\mathcal{NP}$-hard.*

---

[8]Therefore the general result shown in [22] that, given dichotomous preferences, Borda's method and approval voting are equivalent, applies for MMVS as well.

[9]We can show that this result still holds if the voters' preferences are strict orders. I.e., MMVS is also $\mathcal{NP}$-hard if the voters' scoring functions are bijections to $\{1, 2, \ldots, m\}$ and thus no two edges receive the same value.

Proof. Let $q > 2$ be the number of preference classes. Create a graph H from the graph $G = (V, E)$ used in the proof of Theorem 5.1 by concatenating a path p of length $q-2$ to node r. Let $n := |V|$ and $m := |E|$. We now derive from the profile π used in the proof of Theorem 5.1 a profile $\tilde{\pi}$ on the edges of graph H such that $\tilde{\pi}$ consists of q preference classes in two steps. Firstly, we derive from π a preference profile π_1 on G such that every voter i who disapproves of three edges in π is in π_1 replaced by three voters who disapprove of two edges only. Secondly, using the profile π_1 and path p, we assign the edges of H to the preference classes.
In order to get π_1, a voter γ who disapproves of edges $\{\varepsilon_1, \varepsilon_2, \varepsilon_3\}$ is replaced by the following three voters: voter γ_1 who disapproves of edges $\{\varepsilon_1, \varepsilon_2\}$, voter γ_2 who disapproves of edges $\{\varepsilon_2, \varepsilon_3\}$ and voter γ_3 who disapproves of edges $\{\varepsilon_1, \varepsilon_3\}$.
Denote the preference classes that make up $\tilde{\pi}$ by A_{ij}, $0 \leq j \leq q-1$, for all voters i, $1 \leq i \leq k$. Let these preference classes be such that each edge in A_{ij} be strictly preferred to each edge in $A_{ij'}$ for $0 \leq j' < j \leq q-1$. Now for each voter i let $A_{i0} := \{e \in E | a_i(e) = 0\}$ and let $A_{i(q-1)} := \{e \in E | a_i(e) = 1\}$ according to π_1. Note that $|A_{i0}| = 2$ and $|A_{i(q-1)}| = m-2$. Assign the $q-2$ edges of the path p to the classes A_{ij}, $1 \leq j \leq q-2$, in an arbitrary way such that each of these classes contains exactly one edge. Assume Borda voting is being used. Then for every i, voter i's Borda values of the edges are given as follows:

$$b_i(e) = \begin{cases} 2q + (m-3) & \text{if } e \in A_{i(q-1)} \\ 2(j+1) & \text{if } e \in A_{ij},\ 1 \leq j \leq q-2 \\ 1 & \text{if } e \in A_{i0} \end{cases}$$

Obviously each edge of the path p must be contained in a spanning tree of H. Since $2q+(m-3) > 2$ the following two decision problems (D1) and (D2) are equivalent:

(D1)	GIVEN:	Graph G and preference profile π.
	QUESTION:	Is there a spanning tree T of G such that $a_i(T) \geq n-2$ for all i, $1 \leq i \leq k$?
(D2)	GIVEN:	Graph H and preference profile $\tilde{\pi}$.
	QUESTION:	Is there a spanning tree T_1 of H such that $b_i(T_1) \geq (n-2)(2q+(m-3)) + \sum_{j=1}^{q-2} 2(j+1)$ for all i, $1 \leq i \leq k$?

Thus, the corollary follows. □

5.2 Vote-against-t elections and choose-t elections

As a consequence of the proof of Theorem 5.1 in the previous subsection, for any integer $t \geq 2$ MMVS under vote-against-t elections is $\mathcal{NP}$-hard as well. The proof of this result uses the same approach as the one of Theorem 5.1 and is therefore omitted in this paper.

Corollary 5.4 *Let $t \in \mathbb{N}$, $t \geq 2$. Under vote-against-t elections MMVS is $\mathcal{NP}$-hard.*

It is worth noting that the above corollary does not hold for MMVS under vote-against-1 elections. In this case a solution of MMVS can be found in the following way: Remove from the considered graph G all edges e that have $v_i(e) = 0$ for at least one voter i. If the remaining graph is connected, then the objective function value is $n-1$, otherwise it is $n-2$. This observation yields the following statement.

Proposition 5.5 *Under vote-against-1 elections MMVS can be solved in $\mathcal{O}(mk)$ time.*

From Proposition 4.3 we know that MMVS under plurality voting, i.e., choose-1 elections, can be solved within the polynomial time bound of $\mathcal{O}(mk)$. By a reduction from the classical 3SAT problem we can show that, in contrast, MMVS under choose-t elections is $\mathcal{NP}$-hard for each fixed $t \geq 2$. Therefore, as for vote-against-t elections, with the step from $t = 1$ to $t = 2$ the computational complexity of MMVS under choose-t elections jumps from polynomial time solvable to $\mathcal{NP}$-hard.

Theorem 5.6 *MMVS under choose-t elections is $\mathcal{NP}$-hard for every fixed $t \geq 2$.*

6 Conclusion

We have considered the maximin voter satisfaction problem under both the scenarios that the number of voters is constant and may vary. It is known from Aissi et al. [1] that MMVS is polynomially solvable when the number of voters is fixed. The main contribution of this paper has dealt with the question of computational complexity of MMVS in the case of a *variable number of voters*. We improve upon an $\mathcal{NP}$-hardness result of Kouvelis and Yu [13] for general scoring functions by showing that, for a varying number of voters, MMVS is $\mathcal{NP}$-hard under very basic voting rules already. In particular, we have shown that MMVS is computationally intractable under approval voting, vote-against-t elections and choose-t elections for $t \geq 2$. We have proven that the problem is $\mathcal{NP}$-hard both in the cases of dichotomous voter preferences and multichotomous voter preferences. Furthermore, MMVS under Borda voting is $\mathcal{NP}$-hard, irrespective of the underlying voter preferences constituting weak orders or strict orders on the set of edges. Among the voting methods under consideration MMVS has turned out to be polynomially solvable only for the structurally most simple ones: plurality voting and vote-against-1 elections. Thus, when allowing each voter to approve or disapprove of more than one edge, the computational complexity of MMVS jumps from polynomial time solvable to $\mathcal{NP}$-hard. In these $\mathcal{NP}$-hard cases however, it is natural to ask if MMVS is fixed-parameter tractable when parametrized by the number of voters. Following the approach of Aissi et al. [1], we can show that MMVS is fixed-parameter tractable under choose-t elections and under vote-against-t elections, for each $t \geq 2$. Whether or not MMVS under Borda voting is fixed-parameter tractable remains an interesting open question.

References

[1] H. Aissi, C. Bazgan, and D. Vanderpooten. Approximating min-max (regret) versions of some polynomial problems. In D.Z. Chen and D.T. Lee, editors, *COCOON*, volume 4112 of *Lecture Notes in Computer Science*, pages 428–438, Taipei, 2006. Springer, Berlin.

[2] S. Barbera, W. Bossert, and P.K. Pattanaik. Ranking sets of objects. In *Handbook of Utility Theory*, volume 2, pages 893–977. Springer, Berlin, 2004.

[3] C.G. Bird. On cost allocation for a spanning tree: a game theoretic approach. *Networks*, 6:335–350, 1976.

[4] A. Bogomolnaia and H. Moulin. Sharing the cost of a minimal cost spanning tree: beyond the folk solution. *Games and Economic Behavior*, 69(2):238–248, 2010.

[5] S.J. Brams and P.C. Fishburn. *Approval Voting*. Birkhäuser, Boston, 1983.

[6] S.J. Brams and P.C. Fishburn. Voting procedures. In K.J. Arrow, A.K. Sen, and K. Suzumura, editors, *Handbook of Social Choice and Welfare*, volume 1, pages 173–236. Elsevier, Amsterdam, 2002.

[7] S.J. Brams and A.D. Taylor. *Fair Division: From cake-cutting to dispute resolution*. Cambridge University Press, Cambridge, 1996.

[8] A. Darmann, C. Klamler, and U. Pferschy. Finding socially best spanning trees. To appear in *Theory and Decision*.

[9] A. Darmann, C. Klamler, and U. Pferschy. Maximizing the minimum voter satisfaction on spanning trees. *Mathematical Social Sciences*, 58:238–250, 2009.

[10] A. Darmann, C. Klamler, and U. Pferschy. A note on maximizing the minimum voter satisfaction on spanning trees. *Mathematical Social Sciences*, 60:82–85, 2010.

[11] B. Dutta and A. Kar. Cost monotonicity, consistency and minimum cost spanning tree games. *Games and Economic Behavior*, 48:223–248, 2004.

[12] A. Kar. Axiomatization of the shapley value on minimum cost spanning tree games. *Games and Economic Behavior*, 38(2):265–277, 2002.

[13] P. Kouvelis and G. Yu. *Robust discrete optimization and its applications*. Kluwer Academic Publishers, Boston, 1997.

[14] H. Nurmi. *Voting Procedures under Uncertainty*. Springer, Berlin, 2007.

[15] H. Peters, S. Roy, and T. Storcken. Manipulation under k-approval scoring rules. METEOR Research Memorandum 08/056, University of Maastricht, 2008.

[16] J. Rawls. *A Theory of Justice*. Clarendon Press, Oxford, 1972.

[17] F.S. Roberts. *Measurement theory, with applications to Decision Making, Utility and the Social Sciences*. Addison-Wesley, Boston, 1979.

[18] F.S. Roberts. Characterizations of the plurality function. *Mathematical Social Sciences*, 21(2):101–127, April 1991.

[19] D.G. Saari. *Basic Geometry of Voting*. Springer, Berlin, 1995.

[20] T.J. Schaefer. The complexity of satisfiability problems. In *STOC '78: Proceedings of the 10th annual ACM symposium on Theory of Computing*, pages 216–226, San Diego, 1978. ACM, New York.

[21] W. Thomson. Fair allocation rules. Working Paper No.539, 2007. University of Rochester.

[22] M. Vorsatz. Scoring rules on dichotomous preferences. *Social Choice and Welfare*, 31(1):151–162, 2008.

Andreas Darmann, Christian Klamler
Institute of Public Economics
University of Graz
8010 Graz, Austria
Email: {andreas.darmann,christian.klamler}@uni-graz.at

Ulrich Pferschy
Institute of Statistics and Operations Research
University of Graz
8010 Graz, Austria
Email: pferschy@uni-graz.at