Wahlbetrug mit festen und variablen Präferenzen: Eine Komplexitätsanalyse von Kontroll- und Bestechungsproblemen

Inaugural-Dissertation

zur Erlangung des Doktorgrades
der Mathematisch-Naturwissenschaftlichen Fakultät
der Heinrich-Heine-Universität Düsseldorf

vorgelegt von

Cynthia Maushagen
aus Düsseldorf

Düsseldorf, Februar 2021

aus dem Institut für Informatik
der Heinrich-Heine-Universität Düsseldorf

Gedruckt mit der Genehmigung der
Mathematisch-Naturwissenschaftlichen Fakultät der
Heinrich-Heine-Universität Düsseldorf

Berichterstatter:

1. Prof. Dr. Jörg Rothe

2. Jun-Prof. Dr. Dorothea Baumeister

Tag der mündlichen Prüfung: 15.04.2021

ZUSAMMENFASSUNG

In der vorliegenden Dissertation werden verschiedene Kontroll- und Bestechungsprobleme mit Hilfe der klassischen und parametrisierten Komplexitätstheorie untersucht, die sich aus dem Kontext von Wahlen ergeben. Dabei zeichnet sich ein Kontrollproblem dadurch aus, dass der Wahlleiter einer Wahl deren Ergebnis durch Veränderung der Wahlstruktur ändern möchte. Bei Bestechungsproblemen wiederum geht man, wie zu vermuten ist, davon aus, dass einzelne Wähler mittels Bestechung durch einen externen Agenten dazu gebracht werden, ihre Stimme nach dem Willen des Agenten zu ändern. Bei Bestechungsproblemen sind die Präferenzen also variabel, während diese bei Kontrollproblemen fest sind und sich lediglich die Struktur der Wahl verändern lässt. Für die Wahlregeln Veto und Maximin untersuchen wir mit Hilfe der klassischen Komplexitätstheorie zwei Gruppen von Kontrollproblemen. Während in der ersten Gruppe Probleme zur Partitionierung der Wählermenge betrachtet werden, werden in der zweiten Gruppe Probleme zur Partitionierung der Kandidatenmenge betrachtet. Je nach Gruppe teilt der Wahlleiter die Wähler bzw. Kandidaten in zwei Gruppen auf. Um als Sieger aus einer solchen Wahl hervorzugehen, muss sich ein Kandidat in einer Vorrunde und einem Finale gegen die Gewinner der anderen Vorrunde bewähren. Wir zeigen, dass die untersuchten Kontrollprobleme für die Wahlregel Maximin NP-vollständig sind. Für Veto teilen sich die Ergebnisse in NP-Vollständigkeitsresultate und effiziente Algorithmen auf.
Danach widmen wir uns der Wahlregel Plurality und betrachten ein Kontrollproblem aus Sicht der parametrisierten Komplexitätstheorie. Bei diesem Kontrollproblem nehmen wir an, dass der Wahlleiter die Möglichkeit hat, weitere Kandidaten an der Wahl teilnehmen zu lassen. Wir analysieren dazu einen fehlerhaften Beweis aus der Literatur zur W[1]-Härte und können mit Hilfe eines neuen Beweises die W[1]-Härte nachweisen.
Im Anschluss untersuchen wir für insgesamt acht iterative Wahlregeln das Bestechungsproblem Shift-Bribery. Bei diesem Problem gehen wir davon aus, dass der eingangs erwähnte Agent mit einem gewissen Budget zur Bestechung der Wähler ausgestattet ist. Dabei darf der Agent nur die Position eines ausgewählten Kandidaten relativ zu den anderen Kandidaten verändern. Die Kosten für die Veränderung der Stimme eines Wähler ergeben sich hierbei über einen individuell festgelegten Preis, der für jede paarweise Vertauschung zweier Kandidaten veranschlagt wird. Dabei stellt sich heraus, dass dieses Problem für alle untersuchten Wahlregeln NP-vollständig ist.

INHALTSVERZEICHNIS

1 EINLEITUNG

In einer Gesellschaft wird das Leben immer direkt oder mindestens indirekt von Wahlen geprägt. Die Wahl über das künftige Reiseziel innerhalb der Familie, die Wahl für einen Arbeitnehmervertreter im Beruf und die Wahl der Abgeordneten im Bundestag, welche später wieder stellvertretend für uns wählen, stellen nur eine kleine Auswahl an Wahlen dar, die unser Leben beeinflussen. Durch diese Bedeutung, die Wahlen für unser Leben haben, ist es von besonderer Relevanz, dass jeglicher manipulative Einfluss von außen verhindert wird, um eine gerechte Wahl zu garantieren. Angriffe auf Wahlen können dabei nicht nur von außen stattfinden. Bereits das Design der Wahlstruktur birgt die Gefahr, ein Ungleichgewicht in den Chancen für den Sieg einzelner Kandidaten bei einer Wahl zu produzieren. Ein bekanntes Beispiel dafür ist das sogenannte Gerrymandering. In regelmäßigen Abständen wird in den deutschen Medien über Gerrymandering in den Vereinigten Staaten berichtet, so z.B. von Heinzelmann [36], Middelhoff [58], Misteli [59], Steffens [71] und Müller [60].
Das Problem des Gerrymandering ist aber durchaus auch ein Problem in Deutschland. So berichtet beispielsweise Riemenschneider in der Zeitung *Westfälische Nachrichten* [66], dass im schwarz-gelb geführten Nordrhein-Westfalen für die nächste Landtagswahl 2022 die Wahlkreise in und um Münster neu zugeschnitten werden müssen. Die aktuellen Pläne sehen dabei vor, das besonders grün geprägte Zentrum Münsters auf drei Wahlkreise aufzuteilen. Die so neu entstandenen Wahlkreise würden dabei auch jeweils das ländlichere Umland Münsters mit beinhalten. Damit wäre es sehr wahrscheinlich, dass alle drei Wahlkreise an die CDU gehen und das grün geprägte Zentrum keine Gefahr für die CDU in einem der Wahlkreise darstellt. Dadurch, dass es in Deutschland kein direktes Mehrheitswahlsystem gibt, ist der Einfluss von Gerrymandering sicherlich weniger schwerwiegend als in den Vereinigten Staaten. Trotzdem ist dies kein zu vernachlässigendes Thema. Besondere Vorsicht ist geboten, je nachdem, wie eine neue Regelung für die Überhangmandate ausfällt.

GLIEDERUNG DER ARBEIT

Diese Arbeit umfasst Ergebnisse zur Komplexität von Problemen aus der Wahlkontrolle und der Bestechung für verschiedene Wahlregeln. Kapitel 2 liefert uns eine Grundlage für das Verständnis der Komplexitätstheorie. In Kapitel 3 umreißen wir grob die historische Geschichte der Sozialwahltheorie. Kapitel 4 ist in drei Teile gegliedert. Im ersten

Teil wiederholen wir Definitionen und Grundlagen bezüglich Wahlen in der Computational Social Choice und betrachten das Problem der Manipulation und der Gewinnerbestimmung. Im zweiten und dritten Teil befassen wir uns mit der Wahlkontrolle bzw. Bestechung. Dabei stellen wir die Modelle vor, welche wir in einem späteren Teil der Arbeit untersuchen. Zusätzlich schauen wir uns noch weitere Modelle an, welche in der Literatur untersucht werden. Kapitel 5 und Kapitel 6 umfassen eine Komplexitätsanalyse zu verschiedenen Problemen der Wahlkontrolle. Genauer werden wir in Kapitel 5 die Wahlregeln Plurality und Veto betrachten und in Kapitel 6 die Wahlregel Maximin untersuchen. In Kapitel 7 analysieren wir das Problem Shift-Bribery hinsichtlich iterativer Wahlregeln. Schließlich resümieren wir in Kapitel 8 die Ergebnisse dieser Arbeit und richten unseren Blick auf mögliche Fortsetzungen in der Forschung.

2 KOMPLEXITÄTSTHEORIE

Das Ziel dieser Arbeit ist es, Entscheidungsprobleme, die aus dem Kontext der Sozialwahltheorie kommen, hinsichtlich ihrer Komplexität zu klassifizieren. In diesem Kapitel werden wir dafür die grundlegenden Konzepte der klassischen und parametrisierten Komplexitätstheorie wiederholen. Präzise Definitionen und eine umfassende Einführung in die klassische Komplexitätstheorie sind in den Büchern von Rothe [67] und Papadimitriou [63] zu finden. Für eine Einführung in die parametrisierte Komplexitätstheorie sind die Bücher von Cygan et al. [17] und Gurski et al. [35] zu empfehlen.

2.1 KLASSISCHE KOMPLEXITÄTSTHEORIE

Hat man es mit einem algorithmisch lösbaren Problem zu tun, so gibt es im Allgemeinen mehrere sinnvolle Algorithmen, die dieses Problem lösen. Dies bringt uns zu der natürlichen Frage, wie man Algorithmen miteinander vergleichen kann. Um einen sinnvollen Vergleich führen zu können, muss zunächst entschieden werden, welche Parameter miteinander verglichen werden sollen. Zwei besonders interessante Parameter sind der Speicherbedarf und der Zeitverbrauch. Im Rahmen dieser Arbeit konzentrieren wir uns ausschließlich auf den Zeitverbrauch. Um nun Algorithmen hinsichtlich des Zeitverbrauchs vergleichen zu können, muss erst das grundlegende Problem geklärt werden, wie die Zeit, die während der Anwendung eines Algorithmus auf eine vorgegebene Probleminstanz vergeht, gemessen werden soll. Es wäre wenig sinnvoll, die tatsächlich vergangene physikalische Zeit zu messen, die bei der Ausführung einer konkreten Implementierung des Algorithmus vergeht. Durch die enorme Diversität von Rechnerarchitekturen und Programmiersprachen, hätte ein so konstruierter Begriff des Zeitverbrauchs nur wenig Aussagekraft. Stattdessen wird die Zeit abstrakt in der Anzahl der durchzuführenden, elementaren Rechenoperationen gemessen. So können wir Algorithmen bzgl. einzelner Instanzen miteinander vergleichen. Nun kommt es in der Regel vor, dass bei verschiedenen Instanzen mal der eine Algorithmus und mal der andere Algorithmus schneller ist, bzw. weniger Rechenschritte benötigt. Dies trifft vor allem dann zu, wenn Algorithmen für spezielle Typen von Probleminstanzen optimiert wurden. Diese Beobachtung legt nahe, dass es im Allgemeinen nicht ausreicht den Zeitverbrauch zweier Algorithmen auf einer endlichen Menge von Probleminstanzen zu vergleichen. Auf der anderen Seite ist es völlig praxisfern, dass einem alle Laufzeiten bekannt sind, welche bei Anwendung eines

Algorithmus auf eine unendliche und repräsentative Teilmenge aller Probleminstanzen anfallen. An dieser Stelle stattet uns die Komplexitätstheorie mit mehreren Möglichkeiten aus um mit dieser Sachlage umzugehen. Eine solche Möglichkeit besteht in der Betrachtung des sogenannten *worst-case* Szenarios, um welches es in dieser Arbeit geht. Konkret bedeutet dies, dass wir für eine feste Eingabelänge n die Anzahl der Rechenoperationen, welche im worst-case, also bei den am schwersten zu berechnenden Instanzen der Eingabelänge n, benötigt werden, nach oben abschätzen. Für das Wachstum der benötigten Rechenoperationen im worst-case Szenario wird eine Funktion $f\colon \mathbb{N} \to \mathbb{N}$ gesucht, welche dieses Wachstum möglichst genau nach oben beschränkt. Haben wir für zwei Algorithmen solche Funktionen, so schauen wir welche Funktion für immer größer werdende Eingaben weniger stark wächst. Die präzisen Werte $f(n)$ sind hierbei nicht von Bedeutung. Wichtig ist jedoch, dass $f(n)$ zumindest so klein gewählt wird, dass sich der tatsächliche worst-case der Länge n im Sinne der $\mathcal{O}$-Notation nicht von f unterscheidet. In diesem Fall bezeichnen wir f auch als *worst-case-Funktion* des Algorithmus. Diese dient uns als Maß, mit dem wir verschiedene Algorithmen miteinander vergleichen können. Um nun die Frage nach der Vergleichbarkeit zweier Algorithmen A und B zu beantworten, ordnen wir ihnen wie eben beschrieben ihre worst-case-Funktionen f_A bzw. f_B zu und vergleichen anschließend deren asymptotisches Wachstum mittels der $\mathcal{O}$-Notation. Es sei an dieser Stelle jedoch darauf hingewiesen, dass es durchaus noch weitere sinnvolle Möglichkeiten gibt, Algorithmen einem aussagekräftigen Vergleich zu unterziehen. So verrät uns das sogenannte *average-case* Szenario, wie sich die Laufzeit eines Algorithmus im durchschnittlichen Fall verhält.

Neben einem Mittel zum Vergleich von Algorithmen stellen worst-case-Funktionen auch Werkzeuge zur Einteilung algorithmisch zugänglicher Probleme in Komplexitätsklassen dar. Eine besonders wichtige solche Klasse ist die Klasse P aller Entscheidungsprobleme für deren Lösung ein Algorithmus existiert, dessen worst-case-Funktion durch ein Polynom nach oben abgeschätzt werden kann. In diesem Fall sagen wir, dass der Algorithmus eine *polynomielle Laufzeit* hat und das zugehörige Problem *effizient lösbar* ist.

Um die Zugehörigkeit eines Problems zur Klasse P nachzuweisen, reicht es also irgendeinen Algorithmus mit polynomieller Laufzeit anzugeben, der dieses Problem löst. Findet man jedoch keinen Algorithmus mit entsprechender Laufzeit, so kann oft nur schwer ausgeschlossen werden, dass lediglich noch kein solcher Algorithmus gefunden wurde. Es gibt bis heute (Stand Februar 2021) keine Möglichkeit zu beweisen, dass bestimmte Probleme nicht effizient lösbar sind. Eine Möglichkeit um dennoch eine gewisse Sicherheit dafür zu gewährleisten, dass ein Problem nicht in P liegt, erhalten wir aus der Betrachtung polynomieller Reduktionen und der sogenannten Klasse

NP. Bei dieser handelt es sich um die Klasse aller Entscheidungsprobleme für die ein *nichtdeterministischer Algorithmus* existiert, welcher sie in polynomieller Zeit löst. Wir wollen nun das Konzept, welches einem nichtdeterministischen Algorithmus zugrunde liegt anhand eines Bildes erläutern:
Angenommen wir finden uns in einem Labyrinth wieder aus welchem wir den Ausgang finden sollen. Ein algorithmischer Ansatz, sich dieser Aufgabe zu stellen, könnte beispielsweise wie folgt aussehen. Zu jedem Zeitpunkt markieren oder merken wir uns den Weg, den wir bereits abgelaufen sind. Stehen wir vor einer Verzweigung im Labyrinth, so wählen wir den am weitesten links liegenden Weg, welchen wir noch nicht probiert haben. Kommen wir in einer Sackgasse an oder treffen auf ein Wegstück, welches bereits markiert ist, so gehen wir bis zur letzten Verzweigung zurück bei der wir noch nicht alle Pfade ausprobiert haben. Hier wählen wir wieder den am weitesten links liegenden Weg, welchen wir noch nicht genommen haben. Es lässt sich leicht zeigen, dass wir mit dieser Strategie stets den Ausgang finden. Im schlimmsten Fall müssen wir dabei jedoch das komplette Labyrinth durchlaufen bis wir es wieder verlassen können. Nichtsdestotrotz wird uns zu jedem Zeitpunkt genau eine klare Anweisung gegeben, sodass es sich bei diesem Verfahren um einen (deterministischen) Algorithmus handelt. Diese Eindeutigkeit der Handlungsanweisung ist es nun, die bei der Betrachtung nichtdeterministischer Algorithmen verloren geht. Um etwas näher zu erläutern, was damit gemeint ist, bleiben wir im Bilde des Labyrinths und stellen uns vor, dass wir es mit einer ausreichend großen Gruppe von Leuten verlassen wollen. Das Ziel bleibt also, den Ausgang so schnell wie möglich zu finden. Im nichtdeterministischen Fall reicht es jedoch, dass irgendeine Person den Ausgang findet. Dieser wird hier am schnellsten gefunden, wenn sich die Gruppe bei jeder Verzweigung aufteilt, sodass jede Verzweigung gleichzeitig untersucht werden kann. Ist die Gruppe hinreichend groß, dann gibt es mindestens ein Mitglied, welches den Ausgang ohne Umwege über einen Pfad kürzester Länge erreicht. Auch wenn die meisten anderen Mitglieder noch nicht am Ausgang angelangt sind, wird bei einem nichtdeterministischen Algorithmus festgelegt, dass dieser beendet ist, sobald eine Person das Labyrinth verlässt. Diese Festlegung kann dahingehend interpretiert werden, dass ein nichtdeterministischer Algorithmus verschiedene Handlungsoptionen gleichzeitig testen kann. Um die Laufzeit eines nichtdeterministischen Algorithmus zu bestimmen, betrachtet man für jede Eingabe immer den Handlungsstrang, welcher am schnellsten zur Lösung führt. Es sei noch angemerkt, dass jeder deterministische Algorithmus auch als nichtdeterministischer Algorithmus aufgefasst werden kann, da Verzweigungen und parallele Berechnungen der unterschiedlichen Pfade für einen nichtdeterministischen Algorithmus nicht verpflichtend sind.

Als besonders interessant gelten die Probleme in NP, welche vermutlich nicht in P liegen. Cook [16] und Levin [45] zeigten, dass sich solche Probleme in eine tiefere Struktur eingliedern lassen. Um ihr Resultat zu verstehen, müssen wir uns zunächst noch das Konzept der Reduktion von Problemen anschauen. Dieses wollen wir mit Hilfe der beiden Graphprobleme VERTEX COVER und INDEPENDENT SET demonstrieren. Wie üblich verstehen wir unter einem *Graphen* $\mathcal{G}$ eine Menge V von *Knoten* zusammen mit einer Teilmenge E aller 2-elementiger Teilmengen von V, welche wir als die *Kanten* von $\mathcal{G}$ bezeichnen. Im Folgenden betrachten wir zunächst die beiden Probleme einzeln und erläutern anschließend am Beispiel der Reduktion von VERTEX COVER auf INDEPENDENT SET das dahinter liegende Konzept. Schauen wir uns zunächst das Problem VERTEX COVER an, welches als Entscheidungsproblem wie folgt definiert ist.

VERTEX COVER	
Gegeben:	Ein Graph $\mathcal{G} = (V, E)$ und eine natürliche Zahl k.
Frage:	Gibt es eine k-elementige Teilmenge $V' \subset V$, welche die Kantenmenge überdeckt?

Dabei sagen wir, dass eine Kante $\{u, v\}$ von einer Teilmenge $V' \subset V$ überdeckt wird, wenn u oder v in V' enthalten ist. Im folgenden Beispiel untersuchen wir das Problem VERTEX COVER für einen kleinen Graphen $\mathcal{G}$ und verschiedene Parameter k.

Beispiel. *Wir schauen uns den Graph $\mathcal{G}$ mit der Knotenmenge $V = \{v_1, \cdots, v_6\}$ und der Kantenmenge*

$$E = \{\{v_1, v_2\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}, \{v_2, v_5\}, \{v_3, v_5\}, \{v_4, v_5\}, \{v_5, v_6\}\}$$

an, welcher in Abbildung 2.1 dargestellt wird. Gesucht wird eine Teilmenge

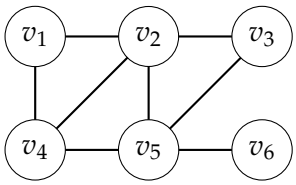

Abb. 2.1: Der Graph $\mathcal{G} = (V, E)$

$V' \subset V$, sodass jede Kante $e \in E$ von V' überdeckt wird. Setzen wir $V' = V$, so gilt offensichtlich, dass unsere Bedingung erfüllt ist und für jede Kante mindestens ein (dann sogar immer genau zwei) Knoten in V' enthalten ist. Schwieriger wird die Aufgabe, wenn die Größe von V' wie in VERTEX COVER *durch eine Zahl k, etwa durch 2, beschränkt wird. In diesem Fall stellt die Teilmenge $V' = \{v_2, v_5\}$ beispielsweise keine Überdeckung von E dar, wie wir an der Kante $\{v_1, v_4\}$ sehen können. Die Abbildung 2.2 zeigt den Graphen $\mathcal{G}$*

mit den markierten Knoten v_2 und v_5 sowie der markierten Kante $\{v_1, v_4\}$. Des Weiteren lässt sich leicht zeigen, dass überhaupt keine 2-elementige Teilmenge $V' \subset V$ existiert, welche E überdeckt. Das Entscheidungsproblem Vertex Cover *kann für die Instanz $(\mathcal{G}, 2)$ also mit nein beantwortet werden.*

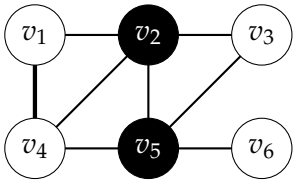

Abb. 2.2: Der Graph $\mathcal{G}$ mit markierten Knoten v_2, v_5 und markierter Kante $\{v_1, v_4\}$

Anders sieht es aus, wenn wir einen der beiden Knoten v_1 oder v_4 zur Menge V' hinzufügen. Dann sehen wir nämlich an den Abbildungen 2.3 und 2.4, dass die Mengen $\{v_1, v_2, v_5\}$ und $\{v_2, v_4, v_5\}$ jeweils eine Überdeckung von E darstellen.

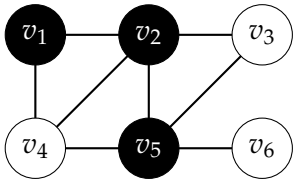

Abb. 2.3: Der Graph $\mathcal{G}$ mit markierten Knoten v_1, v_2 und v_5

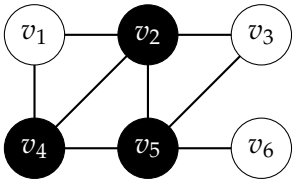

Abb. 2.4: Der Graph $\mathcal{G}$ mit markierten Knoten v_2, v_4 und v_5

Insgesamt können wir für den Graphen $\mathcal{G}$ zusammenfassen, dass das Tupel $(\mathcal{G}, k)$ mit $k < 3$ eine Nein-Instanz und mit $k \geq 3$ eine Ja-Instanz für das Entscheidungsproblem Vertex Cover *ist.*

Wir wollen uns nun das zweitgenannte Problem Independent Set anschauen. Dafür brauchen wir zunächst den Begriff der Unabhängigkeit von Knoten. Wir nennen zwei Knoten eines Graphen *unabhängig voneinander*, wenn sie durch keine Kante dieses Graphen verbunden sind. Das Entscheidungsproblem Independent Set wird formal wie folgt definiert.

Independent Set	
Gegeben:	Ein Graph $G = (V, E)$ und eine natürliche Zahl k.
Frage:	Gibt es eine k-elementige Teilmenge $V' \subset V$, sodass die Knoten in V' paarweise unabhängig voneinander sind?

Anders als bei Vertex Cover, liegt die Schwierigkeit bei Independent Set darin, eine möglichst große Teilmenge $V' \subset V$ zu finden, welche den Anforderungen genügt. Im Folgenden schauen wir uns noch ein Beispiel zu Independent Set an.

Beispiel. *Wir betrachten hier wieder den oben definierten Graphen $\mathcal{G}$. Per Definition bildet jeder einzelne Knoten eine unabhängige Menge. Wir schauen*

uns nun an, wie dies für größere Teilmengen $V' \subset V$ aussieht. Wählen wir den Knoten v_2, so können wir nur noch den Knoten v_6 hinzunehmen um weiterhin eine unabhängige Menge zu erhalten. Ähnlich verhält es sich, wenn wir den Knoten v_5 zuerst wählen. Um eine größere unabhängige Menge zu kriegen, können wir hier nur noch den Knoten v_1 hinzufügen. Ausgehend davon können wir leicht feststellen, dass die einzigen unabhängigen, 3-elementigen Knotenmengen durch $\{v_1, v_3, v_6\}$ und $\{v_3, v_4, v_6\}$ gegeben sind und es keine unabhängige, 4-elementige Knotenmenge gibt. Jede Teilmenge von diesen 3-elementigen Teilmengen bildet wiederum eine unabhängige Menge.

Betrachten wir nun noch einmal die beiden Beispiele nebeneinander. Dann fällt schnell auf, dass die Komplemente der beiden kleinsten überdeckenden Mengen $\{v_1, v_2, v_5\}$ und $\{v_2, v_4, v_5\}$ genau mit den beiden größten, unabhängigen Mengen $\{v_2, v_3, v_6\}$ und $\{v_1, v_3, v_6\}$ übereinstimmen. Dies können wir verallgemeinern. Für eine überdeckende Menge V' der Größe k eines Graphen $\mathcal{G}$ ist das Komplement $V \setminus V'$ stets eine unabhängige Menge der Größe $|V| - k$. Um das zu beweisen, nehmen wir an $V' \subset V$ sei eine überdeckende Menge eines Graphen $\mathcal{G}$. Dann gilt für jede Kante $\{u, v\} \in E$, dass u oder v in V' enthalten ist. Damit gilt für zwei Knoten $s, w \in V \setminus V'$, dass diese durch keine Kante verbunden sein können. Daraus folgt, dass diese Knoten unabhängig voneinander sind, was zu zeigen war. Ebenso leicht kann man sich auch von der Rückrichtung der Aussage überzeugen. Somit haben wir nun einen starken Zusammenhang zwischen den beiden Problemen VERTEX COVER und INDEPENDENT SET gefunden. Um das Resultat von Cook [16] und Levin [45] zu verstehen, müssen wir uns jetzt noch ein paar technische Details anschauen.
Formal gesehen sind die beiden Probleme VERTEX COVER und INDEPENDENT SET Sprachen über einem Alphabet Σ. Dabei ist ein *Alphabet* als eine nichtleere Menge definiert, deren Elemente wir *Buchstaben* nennen. Weiter bezeichnet Σ^* die Menge aller *Wörter*, welche sich aus dem Alphabet Σ bilden lassen. In dieser Terminologie wird eine Teilmenge von Σ^* auch als *Sprache* über dem Alphabet Σ bezeichnet. Die Wörter der Sprachen VERTEX COVER und INDEPENDENT SET sind nun genau die Ja-Instanzen $(\mathcal{G}, k)$ vom jeweiligen Problem.
Wollen wir für einen Graphen $\mathcal{G} = (V, E)$ entscheiden, ob $(\mathcal{G}, k)$ in der Sprache VERTEX COVER enthalten ist, so können wir durch obige Feststellung alternativ prüfen, ob $(\mathcal{G}, |V| - k)$ in der Sprache INDEPENDENT SET enthalten ist. Da die Umformung eines Wortes $(\mathcal{G}, k)$ zum Wort $(\mathcal{G}, |V| - k)$ offenbar in polynomieller Zeit vonstatten geht, sprechen wir hier von einer *polynomialzeit-beschränkten many-one-Reduktion*.
Allgemeiner sagen wir, dass sich ein Problem A mittels einer Reduktion $f : \Sigma^* \to \Sigma^*$ auf ein weiteres Problem B in Polynomialzeit (many-one-)reduzieren lässt, wenn die Bedingung $x \in A \Leftrightarrow f(x) \in B$ für alle $x \in \Sigma^*$ erfüllt ist und die Funktion f *polynomialzeit-berechenbar* ist. Letzteres bedeutet, dass ein Algorithmus mit polynomieller Laufzeit

existiert, welcher für eine beliebige Eingabe $x \in \Sigma^*$ den Funktionswert $f(x)$ berechnet. Folgt man dem Paradigma, dass Algorithmen mit einer polynomiellen Laufzeit effizient sind und deren Laufzeit damit vernachlässigbar ist, dann kann man sagen, dass ein Problem B mindestens so schwer wie ein Problem A ist, wenn sich A auf B in Polynomialzeit reduzieren lässt. Kommen wir nun zu den beiden Problemen Vertex Cover und Independent Set zurück. Dadurch, dass wir von beiden Problemen auf das jeweils andere Problem reduzieren können, ist jedes der Probleme jeweils mindestens so schwer wie das andere. Somit gelten beide Probleme als gleich schwer. Können wir jedes Problem der Klasse NP im obigen Sinne auf ein Problem A reduzieren, so nennen wir A NP-*hart*. Liegt das Problem A zusätzlich selbst in NP, so wird A als NP-*vollständig* bezeichnet.
Cook [16] und Levin [45] zeigten unabhängig voneinander, dass das Problem Satisfiability NP-vollständig ist. Damit war Satisfiability das erste bekannte NP-harte und damit auch NP-vollständige Problem. Da Cook und Levin bei ihren Reduktionen von beliebigen Problemen aus NP ausgehen mussten, ist deren Konstruktion entsprechend abstrakt. Dank ihres Resultats kann jedoch seitdem auf dieses abstrakte Vorgehen verzichtet werden, wenn es darum geht die NP-Härte eines Problems nachzuweisen. Da die Relation der polynomiellen many-one Reduzierbarkeit transitiv ist, reicht es seitdem aus, beim Beweis der NP-Härte eines Problems B eine Reduktion durchzuführen, bei der man ein beliebiges NP-hartes Problem A auf B reduziert.
Für die oben betrachteten Probleme Vertex Cover und Independent Set bewies Karp [44], dass diese NP-vollständig sind. Eine umfangreiche Liste weiterer NP-vollständiger Probleme ist im Buch von Garey und Johnson [33] zu finden.
Bei der unüberschaubaren Menge aller Probleme in NP und der Vielzahl an Forschern, die sich mit der Optimierung von Algorithmen zu diesen Problemen beschäftigen, scheint es äußerst unwahrscheinlich, dass es sich bei P und NP um die gleichen Problemklassen handelt.

2.2 PARAMETRISIERTE KOMPLEXITÄTSTHEORIE

Hat man es mit einem NP-harten Problem zu tun, so ist dieses je nach Anwendungsfall oft dennoch gut handhabbar. Ist man beispielsweise bei dem Problem Vertex Cover ausschließlich an Überdeckungen der Größe k interessiert, wobei k eine vorher fest gewählte Konstante ist, so zeigen Chen et al. [13], dass es einen Algorithmus mit einer Laufzeit in $\mathcal{O}(kn + 1.274^k)$ gibt, welcher das Problem löst. Insbesondere sehen wir, dass das Problem dann effizient lösbar ist. Wie bei Vertex Cover, kann man bei vielen NP-vollständigen Problemen beobachten, dass die Härte des Problems auf einen gewissen Parameter zurückzuführen ist. Die parametrisierte Komplexitätstheorie beschäftigt sich genau mit

diesem Phänomen und bietet eine feinere Gliederung innerhalb der Klasse der NP-harten Probleme.
Diese Theorie wurde Ende 90er Jahre von Downey und Fellows [20] entwickelt und stattet Probleme, welche bereits aus der klassischen Komplexitätstheorie bekannt sind, mit einem zusätzlichen Parameter aus. So gibt es zu einem Problem aus der klassischen Komplexitätstheorie in der Regel mehrere zugehörige, parametrisierte Probleme. Formal gesehen ist ein parametrisiertes Problem eine Teilmenge $L \subset \Sigma^* \times \mathbb{N}$, welche auch als Sprache bezeichnet wird. Dabei steht die erste Komponente für ein Problem aus der klassischen Komplexitätstheorie. Die zweite Komponente bezeichnen wir als Parameter. Bei dem Studium einer parametrisierten Version eines Entscheidungsproblems stellen wir dem Problemnamen ein "p" voran.
Suchen wir nun einen Algorithmus für ein parametrisiertes Problem, so nehmen wir an, dass der Parameter fest gesetzt, also eine Konstante ist. Finden wir einen Algorithmus, der entscheidet, ob $(x, k) \in \Sigma^* \times \mathbb{N}$ eine Ja-Instanz eines solchen Problems ist, und hat dieser Algorithmus eine Laufzeit in $\mathcal{O}(f(k) \cdot |x|^{\mathcal{O}(1)})$, so nennen wir das Problem *fixed parameter tractable*, kurz FPT. Dabei ist f eine beliebige Funktion, die nur von k abhängt. Während das Problem p-Vertex Cover, mit der Größe der Überdeckung als Parameter, FPT ist, wird für p-Independent Set, mit der Größe der unabhängigen Menge als Parameter, vermutet, dass das Problem nicht FPT ist.
Um nachzuweisen, dass ein parametrisiertes Problem A mindestens so schwer wie ein parametrisiertes Problem B ist, genügt es in der parametrisierten Komplexitätstheorie nicht, einen polynomiellen Algorithmus zu finden, der eine Instanz x des Problems A in eine Instanz x' des Problems B überführt und dabei die Eigenschaft, ob es sich um eine Ja- bzw. Nein-Instanz handelt, beibehält. Stattdessen wird hier eine sogenannte *parametrisierte Reduktion* benötigt. Für zwei Sprachen $A, B \subset \Sigma^* \times \mathbb{N}$ ist eine parametrisierte Reduktion von A nach B ein Algorithmus, der eine Instanz $(x, k) \in A$ auf eine Instanz $(x', k') \in B$ abbildet und die drei folgenden Punkte für zwei feste, berechenbare Funktionen f und g erfüllt:

1. Die Instanz (x, k) liegt genau dann in A, wenn die Instanz (x', k') in B liegt.
2. Es gilt $k' \leq g(k)$.
3. Die Laufzeit des Algorithmus liegt in $\mathcal{O}(f(k) \cdot |x|^{\mathcal{O}(1)})$.

Analog zur klassischen Komplexitätstheorie gelten für die parametrisierte Komplexitätstheorie die folgenden zwei Theoreme.

Theorem. *Wenn eine parametrisierte Reduktion eines Problems A zu einem Problem B existiert und B FPT ist, dann ist auch A FPT.*

Theorem. *Wenn jeweils eine parametrisierte Reduktion eines Problems A auf ein Problem B und von B auf ein Problem C existiert, dann gibt es auch eine parametrisierte Reduktion von A nach C.*

Um ein Gefühl für diese Begriffe zu erhalten, schauen wir uns nun an, warum sich die Reduktion von INDEPENDENT SET auf VERTEX COVER nicht zu einer parametrisierten Reduktion fortsetzen lässt. Dazu erinnern wir uns zunächst daran, dass wir im Rahmen der klassischen Komplexitätstheorie bei der Reduktion von INDEPENDENT SET auf VERTEX COVER, eine INDEPENDENT SET-Instanz $(\mathcal{G}, k)$ auf eine VERTEX COVER-Instanz $(\mathcal{G}, |V| - k)$ geschickt haben. Da der Wert $|V|$ beliebig groß im Vergleich zu k werden kann, hat das zur Folge, dass es keine Funktion g gibt, sodass $|V| - k \leq g(k)$ für alle $k \in \mathbb{N}$ und jegliche Knotenmenge V erfüllt ist. Dadurch ist der 2. Punkt der Definition einer parametrisierten Reduktion nicht erfüllt.
Wie bereits oben angedeutet, wird im Allgemeinen angenommen, dass p-INDEPENDENT SET für den Parameter *Größe der unabhängigen Menge* nicht FPT ist. Neben einem Werkzeug zur Einteilung parametrisierter Probleme in solche die FPT sind und solche die es vermutlich nicht sind, bietet die parametrisierte Komplexitätstheorie für letztere eine Möglichkeit sie in die sogenannte *W-Hierarchie* einzuordnen. Um die W-Hierarchie einzuführen, brauchen wir noch ein paar Definitionen.

Es folgt dafür ein kurzer Abschnitt, in dem wir gerichtete Graphen und einige ihrer Eigenschaften definieren.
Unter einem *gerichteten Graphen* $\mathcal{G}$ verstehen wir ein Tupel (V, E), wobei V eine nichtleere Menge von Knoten und $E \subset V \times V$ eine Menge von *gerichteten Kanten* ist. Sei nun $\mathcal{G} = (V, E)$ ein solcher gerichteter Graph. Wir nennen eine Folge von l Knoten $C = (v_1, \ldots, v_l)$ einen *Weg*, wenn $(v_i, v_{i+1}) \in E$ für alle $i \in \{1, \ldots, l-1\}$ gilt und die Knoten paarweise verschieden sind. Die *Länge eines Weges* $C = (v_1, \ldots, v_l)$ ist definiert als $l - 1$. Ein Weg $C = (v_1, \ldots, v_l)$ wird auch auch als *Kreis* bezeichnet, falls zusätzlich die Kante $(v_l, v_1) \in E$ enthalten ist. Besitzt $\mathcal{G}$ keinen Kreis, so nennen wir $\mathcal{G}$ auch *kreisfrei*. Der *Eingangsgrad* eines Knotens $v \in V$ ist durch $|\{u \in V \mid (u, v) \in E\}|$ definiert. Analog dazu ist der *Ausgangsgrad* durch $|\{u \in V \mid (v, u) \in E\}|$ definiert.

Für die Einführung der W-Hierarchie benötigen wir noch die Definition boolescher Schaltkreise.
Ein *boolescher Schaltkreis* ist ein gerichteter, kreisfreier Graph $\mathcal{G}$, dessen Knoten mit einem konstanten Wahrheitswert (**true** oder **false**), einer Eingabevariablen $x_i \in \{x_1, x_2, \ldots, x_n\}$ oder einer booleschen Operation $\wedge$, $\vee$ und $\neg$ markiert sind. Die Knoten werden entsprechend nach ihrer Markierung **true**-, **false**-, x_i-, $\wedge$-, $\vee$- oder $\neg$-*Gatter* benannt. Die x_i-Gatter, sowie die **true**- und **false**-Gatter haben einen Eingangsgrad 0. Ein x_i-Gatter nennen wir auch *Eingabegatter*. Während die $\neg$-Gatter einen Eingangsgrad von 1 haben, haben die $\vee$- und $\wedge$-Gatter einen Eingangsgrad von mindestens 2. Hat ein Gatter einen Ausgangsgrad von 0, so wird dieses auch als *Ausgabegatter* bezeichnet. Hat ein boolescher

Schaltkreis mehrere Ausgangsgatter so können mehrere boolesche Funktionen gleichzeitig berechnet werden.
Jede Belegung T der Variablen $x_1, x_2, \ldots, x_n$ definiert für jedes Gatter g einen Wahrheitswert $f(g) \in \{\textbf{true}, \textbf{false}\}$, der sich induktiv wie folgt definieren lässt:

- Falls g ein **true**-Gatter, **false**-Gatter bzw. x_i-Gatter ist, so ist $f(g)$ **true**, **false** bzw. $T(x_i)$,
- Ist g ein $\neg$-Gatter und g' der Vorgänger von g, dann ist $f(g) =$ **true**, falls $f(g') =$ **false** ist, und umgekehrt $f(g) =$ **false**, falls $f(g') =$ **true** gilt.
- Ist g ein $\wedge$-Gatter (bzw. ein $\vee$-Gatter) und sind $g_1, \ldots, g_n$ die Vorgänger von g, dann ist $f(g) =$ **true**, falls $f(g_i) =$ **true** für alle $i \in \{1, \ldots, n\}$ (bzw. falls $f(g_i) =$ **true** für ein $i \in \{1, \ldots, n\}$) ist. Andernfalls gilt $f(g) =$ **false**.

Gatter in einem booleschen Schaltkreis, die einen Eingangsgrad größer als zwei haben, werden als *große Gatter* bezeichnet. Die *Weft eines booleschen Schaltkreises* ist die maximale Anzahl an großen Gattern auf einem gerichteten Weg von einem Eingangsgatter zum Ausgangsgatter. Die *Höhe eines booleschen Schaltkreises* ist die Länge eines längsten gerichteten Wegs von einem Eingang zu einem Ausgang.
Durch das wie folgt definierte, parametrisierte Problem p-WEIGHTED CIRCUIT SATISFIABILITY(t, h), können wir endlich die W-Hierarchie definieren.

p-WEIGHTED CIRCUIT SATISFIABILITY(t, h)	
Gegeben:	Ein boolescher Schaltkreis C mit Weft t, Höhe h und eine natürliche Zahl k.
Parameter:	k.
Frage:	Gibt es eine erfüllende Belegung von C mit Gewicht k?

Das *Gewicht* einer erfüllenden Belegung ist dabei als die Anzahl der Variablen definiert, die mit dem Wahrheitswert **true** belegt werden.
Für $t \geq 1$, gehört ein parametrisiertes Problem P zur Klasse $\mathrm{W}[t]$, falls eine parametrisierte Reduktion von P auf das Problem p-WEIGHTED CIRCUIT SATISFIABILITY(t, h) mit $h \geq 1$ existiert.
Die Komplexitätsklassen $\mathrm{W}[t]$ für $t \geq 1$ bilden zusammen die *W-Hierarchie*.
Ein parametrisiertes Problem P heißt *W[t]-hart*, wenn für jedes Problem der Klasse $\mathrm{W}[t]$ eine parametrisierte Reduktion auf P existiert. Liegt P zusätzlich in $\mathrm{W}[t]$, so nennen wir P auch *W[t]-vollständig*.

Das folgende Beispiel zeigt einen booleschen Schaltkreis.

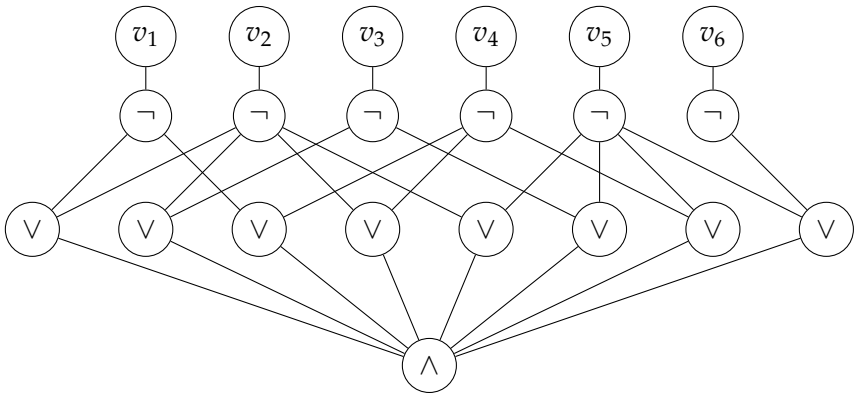

Abb. 2.5: Der boolesche Schaltkreis zur INDEPENDENT SET-Instanz aus Abschnitt 2.1

Beispiel. *Die Abbildung 2.5 zeigt den booleschen Schaltkreis zur* INDEPENDENT SET*-Instanz aus Abschnitt 2.1.*

Der boolesche Schaltkreis besitzt genau ein großes Gatter, welches hier gleichzeitig das Ausgabegatter ist, wodurch sich eine Weft von 1 *ergibt. Die Höhe des booleschen Schaltkreises beträgt hier* 3*.*

Das Beispiel skizziert die Idee, wie eine parametrisierte Reduktion von p-INDEPENDENT SET auf p-WEIGHTED CIRCUIT SATISFIABILITY$(1, 3)$ aussieht. Mit einer solchen Reduktion lässt sich nun zeigen, dass p-INDEPENDENT SET in W[1] liegt. Den Nachweis der W[1]-Härte erbrachten Downey und Fellows [19].

3

SOZIALWAHLTHEORIE

Als Quelle für dieses Kapitel dienten die Bücher von Arrow et al. [2] und Thiele [73].
Die Sozialwahltheorie ist die Theorie der kollektiven Entscheidungen. In einer Gruppe von Menschen müssen immer wieder Entscheidungen getroffen werden. So unterschiedlich verschiedene Menschen sind, so unterschiedlich können auch ihre Präferenzen über verschiedene Wahloptionen sein. Das hat zur Folge, dass eine sinnvolle Aggregation verschiedener Präferenzen in ein Wahlergebnis nicht trivial ist. Dabei reicht die Möglichkeit für das Design einer Wahlregel von der Diktatur, in der ein Einzelner die Macht besitzt alleine zu bestimmen, bis zur Einstimmigkeitsregel, wo ein Kandidat nur gewinnen kann, wenn jeder Wähler für ihn stimmt. Bis ins 12. Jahrhundert konnten im kanonischen Recht bei Personalentscheidungen Wahlen nur durch Einstimmigkeit beschlossen werden, da dies als ein Zeichen göttlichen Willens gewertet wurde. Nun ist es so, dass Diktaturen in liberalen Gesellschaften mehrheitlich abgelehnt werden und die Einstimmigkeitsregel zur Konsequenz hat, dass praktisch niemals ein Beschluss gefasst werden kann. Zwischen diesen beiden sehr extremen Wahlregeln gibt es eine Vielzahl von weiteren Wahlregeln, welche bei Wahlen auf der ganzen Welt verwendet werden.
In der Geschichte der Sozialwahltheorie haben Wissenschaftler aus verschiedenen Disziplinen wie z.B. der Philosophie, Mathematik und Ökonomie immer wieder neue Impulse gesetzt. Dabei sind Sokrates und Platon um 400 v. Chr., über Plinius den Jüngeren im 2. Jahrhundert, über Ramon Llull im 13. Jahrhundert zu Jean-Charles de Borda und Marquis de Condorcet im 18. Jahrhundert einige der prominentesten Vertreter der eben genannten Wissenschaften, die sich diesem Thema widmeten.
Trotz der langen Zeit der Forschung, wählt Frankreich, ein Land welches Deutschland nicht nur geografisch nahe steht, den französischen Präsidenten durch ein anderes Wahlverfahren als wir in Deutschland den deutschen Kanzler. Damit stellt sich die Frage, ob nicht bereits bekannt ist, dass eins der beiden Wahlverfahren besser ist oder ob es nicht ein noch besseres oder sogar perfektes Wahlsystem gäbe. Dafür müsste zuerst geklärt werden, wie man Wahlregeln miteinander vergleicht. Wahlregeln lassen sich hinsichtlich verschiedener Eigenschaften kategorisieren. Um eine Vorstellung davon zu erhalten, wie so eine Eigenschaft aussehen kann, betrachten wir zwei exemplarisch. Eine intuitiv sinnvolle Eigenschaft beschreibt das *Pareto-Kriterium*. Dieses besagt, dass ein Kandidat a nicht zu den Gewinnern der Wahl

gehören darf, wenn es einen Kandidaten b gibt, der von jedem Wähler gegenüber a bevorzugt wird. Eine andere wünschenswerte Eigenschaft beschreibt das Kriterium *Unabhängigkeit von irrelevanten Alternativen*. Damit eine Wahlregel diese Eigenschaft erfüllt muss für eine gegebene Wahl das Folgende gelten. Angenommen an dieser Wahl nehmen die Kandidaten a, b und möglicherweise weitere Kandidaten teil und Kandidat a gewinnt die Wahl. Wiederholen wir nun diese Wahl und lassen einen zusätzlichen Kandidaten c an der Wahl teilnehmen, während alle Wähler ihre alte Präferenz bezüglich der anderen Kandidaten beibehalten und nur den Kandidaten c in ihrer Präferenz einordnen, so garantiert die Unabhängigkeit von irrelevanten Alternativen, dass b kein Gewinner der Wahl ist.
Nun könnte man prüfen, welches Wahlsystem mehr wünschenswerte Eigenschaften erfüllt. Wenn eine Wahlregel A nur eine echte Teilmenge der wünschenswerten Eigenschaften einer Wahlregel B erfüllt, so ist sicherlich die Wahlregel B zu bevorzugen. Wenn die beiden Wahlregeln allerdings unterschiedliche wünschenswerte Eigenschaften erfüllen, dann ist ein Vergleich wohl eher nur subjektiv bewertbar. Ein perfektes Wahlsystem wäre vermutlich eines, welches alle wünschenswerten Eigenschaften erfüllt. Während in der Anfangszeit der Sozialwahltheorie Wahlregeln einzeln auf Eigenschaften geprüft wurden, so wird in der modernen Sozialwahltheorie auch der Zusammenhang verschiedener Eigenschaften unabhängig von einzelnen Wahlregeln untersucht. Die moderne Sozialwahltheorie erfährt als wissenschaftliche Disziplin in der Mitte des 20. Jahrhunderts ihre Anfangsphase. Eines der wichtigsten und zugleich enttäuschendsten Resultate aus dieser Zeit stammt von Arrow [1]. Er bewies, dass eine Wahlregel, welche das Pareto-Kriterium und das Kriterium der Unabhängigkeit von irrelevanten Alternativen erfüllt automatisch eine Diktatur ist. Dieses Resultat ist eines von mehreren sogenannten Unmöglichkeitsresultaten, welche aussagen, dass gewisse wünschenswerte Eigenschaften nur innerhalb einer Diktatur gemeinsam realisiert werden können. Die Frage nach der Existenz eines perfekten Wahlsystems lässt sich daher sicherlich mit nein beantworten. Das hat zur Folge, dass Gesellschaften, die nicht in einer Diktatur leben wollen, stets gezwungen sind positive Eigenschaften einer Wahl derart zu gewichten, dass diese durch ein Wahlsystem realisiert werden können.

4

COMPUTATIONAL SOCIAL CHOICE

Die Computational Social Choice ist ein noch recht junges Forschungsgebiet, welches aus der Sozialwahltheorie und der Informatik entstanden ist. Hierbei ist es vor allem die theoretische Informatik, die eine neue Perspektive auf viele Probleme der Sozialwahltheorie bietet. Ein gutes Beispiel für diese neu gewonnene Perspektive steht im Zusammenhang mit dem Gibbard-Satterthwaite-Theorem. Dieses ist neben Arrows Theorem ein weiteres Unmöglichkeitsresultat, welches das gleichzeitige Auftreten bestimmter wünschenswerter Eigenschaften von Wahlregeln ausschließt. Genauer haben Gibbard [34] und Satterthwaite [69] unabhängig voneinander bewiesen, dass es sich bei einer Wahlregel, bei der potentiell jeder Kandidat gewinnen kann und die nicht manipulierbar ist, um eine Diktatur handelt. Anders als der Begriff Manipulation zunächst vermuten lässt, gilt es in der Sozialwahltheorie bereits als Manipulation, wenn ein Wähler aus strategischen Gründen entgegen seiner eigentlichen Interessen wählt. Konfrontiert mit den Alternativen einer Diktatur und Wahlen, bei denen gewisse Kandidaten vorab als Verlierer feststehen, ist es sicherlich am einfachsten hinnehmbar, wenn Wähler das Wahlergebnis durch strategisches Wählen beeinflussen können.
Trotzdem ist strategisches Wählen natürlich nicht wünschenswert und so wurde ein Ausweg aus der Misere gesucht, welcher schließlich von Bartholdi et al. [3] gefunden wurde. Mit Hilfe der theoretischen Informatik haben sie untersucht, wie schwer es für einen unehrlichen Wähler ist die Wahl in seinem Sinne zu manipulieren. Dabei gingen sie von der Grundannahme aus, dass eine hohe Komplexität, die mit der Manipulation einhergeht, einen ausreichenden Schutz vor einer solchen Manipulation darstellt. Während also in der Sozialwahltheorie abstrakt nach der Existenz von Wahlsystemen mit gewissen Eigenschaften gesucht wird und Verbindungen zwischen diesen Eigenschaften hergeleitet werden, steht in der Computational Social Choice die Komplexität von Problemen im Vordergrund, die mit Wahlen einhergehen.

Das restliche Kapitel ist in drei Teile gegliedert. Im ersten Teil führen wir die für uns relevanten Definitionen zu Wahlen ein und schauen uns die Probleme der Gewinnerbestimmung und der Manipulation an. Im zweiten und dritten Teil beschäftigen wir uns mit der Kontrolle von Wahlen bzw. mit der Bestechung von Wählern, schauen uns verschiedene Modelle an und ordnen diese ein. Wir werden uns dabei stets an die Definitionen und Notationen aus dem Buch von Rothe [68] halten.

4.1 WAHLEN, GEWINNERBESTIMMUNG UND MANIPULATION

Wir beginnen mit der Wiederholung der wichtigsten Definitionen bezüglich Wahlen. Eine *Wahl* E wird durch ein Tupel (C, V) definiert, wobei $C = \{c_1, c_2, \ldots, c_m\}$ eine *Menge von Kandidaten* ist und $V = (v_1, v_2, \ldots, v_n)$ eine *Liste von Präferenzen* über die Kandidaten bezeichnet. Dabei ist eine *Präferenz*, auch *Stimme* genannt, als vollständige lineare Ordnung über der Kandidatenmenge definiert. Wir gehen also davon aus, dass die Präferenz eines jeden Wählers durch eine vollständige Rangordnung über die Kandidaten gegeben ist. Zwecks sprachlicher Vereinfachung identifizieren wir im Folgenden einen Wähler mit seiner Stimme bzw. Präferenz. Eine *Wahlregel*, auch *Wahlsystem* genannt, ist eine Funktion, die jeder Wahl eine Teilmenge ihrer Kandidaten zuordnet, welche wir als die *Gewinner* der Wahl interpretieren. Eine natürliche Familie von Wahlregeln bilden die *Scoring-Protokolle*. Dabei gehört zu jedem Scoring-Protokoll ein aus einer absteigenden Folge nichtnegativer Zahlen bestehender *Scoring-Vektor* $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_m)$. Naturgemäß wird hierbei festgelegt, dass ein Kandidat, der an der i-ten Stelle einer Stimme steht, die Punktzahl α_i erhält. Wie zu erwarten, ist der Gesamtpunktestand eines Kandidaten als die Summe der Punkte definiert, die er über alle Stimmen hinweg erhält. Die Kandidaten mit dem höchsten Punktestand sind die Gewinner der Wahl. Der vielleicht prominenteste Vertreter aus der Familie der Scoring-Protokolle ist *Plurality*. Bei diesem bekommt nur der am meisten präferierte Kandidat eines Wählers einen Punkt und alle weiteren Kandidaten erhalten keinen Punkt. Deutlich mehr Punkte werden bei der Wahlregel *Veto* vergeben. Hier erhält ein Kandidat genau dann einen Punkt durch einen Wähler, wenn er nicht an der letzten Stelle der Stimme des Wählers steht. Das kann dahingehend interpretiert werden, dass jeder Wähler ein Veto an einen der Kandidaten vergeben kann. Zuletzt sei noch die Wahlregel *Borda* erwähnt, die dadurch charakterisiert wird, dass ein Kandidat für die Position i in der Präferenz eines Wählers genau $m - i$ Punkte erhält. Zu den vorgestellten Scoring-Protokollen gehören somit die Scoring-Vektoren $(1, 0, \ldots, 0)$, $(1, \ldots, 1, 0)$ und $(m-1, m-2, \ldots, 1, 0)$.

Mit dem Resultat zur Manipulation von Wahlen haben Bartholdi et al. [3] einen Grundstein für das Forschungsgebiet der Computational Social Choice gelegt. Da die Manipulation von Wahlen eng verwandt mit der Kontrolle und Bestechung von Wahlen ist, wollen wir im Folgenden die Manipulation noch einmal genauer betrachten.

Das folgende Beispiel illustriert, wie Manipulation bei einer gegebenen Wahl aussehen kann.

Beispiel. *Wir betrachten die Wahl* $(\{a, b, c, d\}, V)$ *mit der wie folgt gegebenen Liste von Präferenzen* V*:*

		3	2	1	0
v_1	:	a	$>$ b	$>$ c	$>$ d
v_2	:	a	$>$ c	$>$ d	$>$ b
v_3	:	d	$>$ c	$>$ a	$>$ b
v_4	:	d	$>$ c	$>$ b	$>$ a
v_5	:	c	$>$ a	$>$ b	$>$ d

Die Kandidaten a*,* b*,* c *und* d *erreichen unter Anwendung der Wahlregel Borda* 9*,* 4*,* 10 *bzw.* 7 *Punkte. Wir sehen also, dass* c *durch Anwendung von Borda zum Gewinner der Wahl wird.*
Nehmen wir nun an, dass der Wähler v_2 *in etwa weiß wie die Präferenzen der anderen Wähler aussehen. Dann könnte er statt seiner eigentlichen Präferenz die Stimme* $a > d > b > c$ *abgeben. In diesem Fall bekämen die Kandidaten* a*,* b*,* c *und* d *der Reihenfolge nach* 9*,* 5*,* 8 *und* 8 *Punkte. Insbesondere hätte das zur Folge, dass der von* v_2 *am meist präferierte Kandidat* a *die Wahl gewinnt. Der Wähler* v_2 *kann also durch Veränderung seiner Stimme die Wahl zu seinen Gunsten manipulieren.*

Wie dieses Beispiel zeigt, kann die Manipulation von Wahlen für bestimmte Instanzen ziemlich einfach sein. Um eine Wahlregel zu finden, die nur schwer manipuliert werden kann, untersuchen wir das Entscheidungsproblem ε-MANIPULATION. Die Variable ε steht dabei stets für eine Wahlregel. Das Problem ist formal wie folgt definiert.

ε-MANIPULATION	
Gegeben:	Eine Wahl (C, V) und ein ausgewählter Kandidat $p \in C$.
Frage:	Gibt es eine lineare Präferenz v, sodass p der alleinige Gewinner der Wahl ist, wenn v der Wahl hinzugefügt wird?

Ist bereits die Gewinnerbestimmung bei einer Wahl schwer, d.h. nicht effizient lösbar, dann ist natürlich auch auch die Manipulation schwer. Eine schwere Gewinnerbestimmung kann allerdings nicht wünschenswert sein, da es ein natürliches Interesse der Wähler ist das Wahlergebnis zeitnah zu erhalten. Das Entscheidungsproblem der Gewinnerbestimmung ist dabei wie folgt definiert.

ε-WINNER	
Gegeben:	Eine Wahl (C, V) und ein ausgewählter Kandidat $p \in C$.
Frage:	Ist p unter Verwendung der Wahlregel ε der alleinige Gewinner der Wahl?

Wie man leicht erkennen kann, bilden Scoring-Protokolle eine Klasse von Wahlregeln, für welche die Gewinnerbestimmung effizient berechenbar ist. Hat eine Wahlregel keine effiziente Gewinnerbestimmung, so ist diese für die praktische Anwendung in den meisten Fällen nicht sinnvoll einsetzbar. Das Interesse an der Manipulation von Wahlregeln wie *Dodgson*, bei denen bereits die Gewinnerbestimmung NP-schwer ist, ist daher eher von theoretischer Natur. Den Beweis der NP-Schwere von Dodgson-Winner führten Bartholdi et al. [4]. Auch bei den Problemen der Kontrolle und Bestechung von Wahlen, welche wir in den nächsten Abschnitten betrachten, soll die Schwere der Probleme jeweils nur im Angriff auf die Wahl liegen. In dieser Arbeit werden daher nur Wahlregeln untersucht, bei denen die Gewinnerbestimmung effizient lösbar ist.

Eine weitere Gemeinsamkeit zwischen der Manipulation, Kontrolle und Bestechung liegt in der Annahme, dass einem Angreifer alle Stimmen der ehrlichen Wähler bekannt sind. Wahrscheinlich hat man nur in sehr seltenen Fällen tatsächlich die vollständige Liste der Präferenzen bevor ein solcher Angriff auf die Wahl durchgeführt wird. Wir schauen uns nun verschiedene Gründe an, warum eine solche Annahme trotzdem sinnvoll ist. Den ersten Grund liefern Bartholdi et al. [3] in ihrer Arbeit bereits selbst. Sie verteidigen die Annahme damit, dass eine Manipulation, die bereits schwer ist, obwohl man das Wissen über alle Wählerstimmen hat, nicht einfacher sein kann, wenn dort noch Ungewissheiten dazukommen. Weiter lassen sich durch Wahlprognosen gute Schätzungen modellieren, welche für die Eingabe genutzt werden können und uns als zweite Rechtfertigung dieser Annahme dienen. Betrachtet man die untersuchten Probleme nicht nur als Angriffe auf eine Wahl, so kommt ein weiterer guter Grund hinzu. So lassen sich beispielsweise verschiedene Bestechungsprobleme auch als Messinstrument einsetzen, welches misst ob ein Angriff auf eine bereits ausgewertete Wahl stattgefunden haben könnte. Nach einer Wahl sind die Stimmen bekannt und können dadurch für die Berechnungen tatsächlich genutzt werden.
In der Literatur finden sich aber durchaus auch Arbeiten, in denen davon ausgegangen wird, dass nur ein Teil der Wählerliste bekannt ist. Beispiele hierfür liefern die Arbeiten von Hemaspaandra et al. [39–41], in denen die Probleme der Online Manipulation, Online Kontrolle und Online Bestechung betrachtet werden. Hier ist dem Angreifer immer nur ein Teil der Wählerliste bekannt, sodass er eine Angriffsstrategie finden muss, die unabhängig von den noch restlichen Wählerstimmen funktioniert.
In dieser Arbeit beschränken wir uns auf Probleme bei denen die gesamte Wählerliste gegeben ist.

4.2 KONTROLLE VON WAHLEN

Die Kontrolle von Wahlen ist Forschungsgegenstand in den Arbeiten aus Kapitel 5 und 6.
Unter den Begriff der *Wahlkontrolle* fallen verschiedene Szenarien, in denen beispielsweise ein Wahlleiter durch Veränderungen der Wahlstruktur Einfluss auf den Ausgang der Wahl nimmt. Wir betrachten zunächst die von Bartholdi et al. [5] definierten Kontrollprobleme, bei denen der Wahlleiter durch das

- Hinzufügen oder Löschen von Kandidaten,
- Hinzufügen oder Löschen von Wählern oder
- Einführen einer mehrstufigen Wahl

einen ausgewählten Kandidaten zum Gewinner der Wahl machen möchte. Hemaspaandra et al. [38] definierten die dazu destruktive Variante, bei welcher der Wahlleiter den Sieg eines ausgewählten Kandidaten verhindern möchte.
Im Folgenden betrachten wir die von Bartholdi et al. definierten Probleme, die mit einer mehrstufigen Wahl einhergehen. Durch die Änderung auf eine rundenbasierte bzw. mehrstufige Wahl stehen dem Wahlleiter bei der Gestaltung der Wahl eine Vielzahl an Möglichkeiten offen. Die von Bartholdi et al. definierten Probleme lassen sich in zwei Kategorien einteilen. Bei den Problemen der ersten Kategorie werden die Kandidaten in zwei Gruppen eingeteilt und müssen sich in einer Vorrunde für das Finale qualifizieren. Die Kandidaten dürfen hier in beliebiger Weise aufgeteilt werden. Die Wähler stimmen dabei in den beiden Vorrunden und im Finale gemeinsam ab. Das erste Problem, welches wir betrachten, ist ε-Constructive-Control-by-Run-off-Partition-of-Candidates-TE. Dieses ist für eine beliebige Wahlregel ε wie folgt definiert.

ε-Constructive-Control-by-Run-off-Partition-of-Candidates-TE	
Gegeben:	Eine Wahl (C, V) und ein ausgewählter Kandidat $p \in C$.
Frage:	Gibt es eine Partition von C in C_1 und C_2, sodass p der alleinige Sieger der zweistufigen Wahl ist, in der (bezüglich der Stimmen aus V) die Gewinner der Vorwahlen (C_1, V) und (C_2, V), die in ihrer Vorwahl jeweils die Gleichstandsbrechungsregel TE (ties eliminate) überstehen, gegeneinander antreten?

Ein ähnliches Problem ergibt sich, wenn wir die Gleichstandsbrechungsregel TE durch TP (ties promote) ersetzen. Bei Anwendung dieser Regel dürfen alle Gewinner einer Vorwahl am Finale teilnehmen. Dieses zweite Problem nennt sich ε-Constructive-Control-by-Run-off-Partition-of-Candidates-TP. Eine zusätzliche Variante der

beiden Probleme ergibt sich, wenn es lediglich eine Vorwahl (C_1, V) gibt und die Kandidaten der Gruppe C_2 direkt ins Finale einziehen. Diese Probleme nennen sich ε-Constructive-Control-by-Partition-of-Candidates-TE und ε-Constructive-Control-by-Partition-of-Candidates-TP. Das erinnert an manche Sportveranstaltungen, in denen topgesetzte Spieler oder Teams keine Vorrundenspiele bestreiten müssen. Alle vier definierten Probleme gibt es auch in einer destruktiven Version. Hier möchte der Wahlleiter nun verhindern, dass der ausgewählte Kandidat als alleiniger Sieger aus der Wahl hervorgeht. Dabei ist es bei einer Ja-Instanz ausreichend, dass ein anderer Kandidat ein Gewinner der Wahl wird. Im Namen dieser Probleme ändert sich in diesem Fall das Wort Constructive zu Destructive.
Alle definierten Problem sind im sogenannten *unique-winner* Modell formuliert. Daneben existiert noch das *nonunique-winner* Modell. Dabei ist im konstruktiven Fall das Problem so definiert, dass es für den ausgewählten Kandidaten ausreicht ein Gewinner neben möglicherweise mehreren Gewinnern zu sein. Im destruktiven Fall muss entsprechend verhindert werden, dass der ausgewählte Kandidat zu den Gewinnern der Wahl zählt.
Hemaspaandra et al. [37, Thm. 8, S. 386] haben gezeigt, dass die erzeugten Sprachen zu den Problemen ε-Destructive-Control-by-Run-off-Partition-of-Candidates-TP und ε-Destructive-Control-by-Partition-of-Candidates-TP im nonunique-winner Modell für jede Wahlregel ε identisch sind. Selbiges gilt für die Sprachen zu den beiden Problemen ε-Destructive-Control-by-Run-off-Partition-of-Candidates-TE und ε-Destructive-Control-by-Partition-of-Candidates-TE, welche im unique-winner als auch im nonunique-winner Modell übereinstimmen. Damit umfasst diese erste Kategorie der von uns betrachteten Kontrollprobleme effektiv 13 verschiedene Probleme.

Kommen wir nun zu der zweiten Kategorie, der von uns betrachteten Kontrollprobleme. Bei dieser wird nun die Wählermenge in zwei Wählergruppen aufgeteilt. Beide Wählergruppen ermitteln jeweils unabhängig voneinander die Sieger, der durch sie definierten Vorrunden. Anschließend müssen sich die Gewinner der beiden Vorrunden in einem gemeinsamen Finale dem Voting aller Wähler stellen. Wie schon bei der Partitionierung der Kandidatenmenge, können auch bei der Partitionierung der Wählermenge unterschiedliche Varianten betrachtet werden. Das folgende Entscheidungsproblem ε-Constructive-Control-by-Partition-of-Voters-TE präzisiert eine davon.

ε-Constructive-Control-by-Partition-of-Voters-TE	
Gegeben:	Eine Wahl (C, V) und ein ausgewählter Kandidat $p \in C$.
Frage:	Gibt es eine Partition von V in V_1 und V_2, sodass p der alleinige Sieger der Wahl ist, in der die Gewinner der Vorwahlen (C, V_1) und (C, V_2), die in ihrer Vorwahl jeweils die Gleichstandsbrechungsregel TE überstehen, gegeneinander antreten?

Durch die Alternativen, die sich aus der Betrachtung des destruktiven Falls, der Änderung auf das nonunique-winner Modell oder der Anwendung der Gleichstandsbrechungsregel TP ergeben, erhalten wir insgesamt acht Probleme zur Partitionierung der Wählermenge.
Für reale Anwendungen lassen sich die Probleme der Partitionierung der Wählermenge in der hier angegebenen Form kaum anwenden. Grund dafür ist die hohe Flexibilität, die dem Wahlleiter bei der Partitionierung gewährt wird. So darf beispielsweise die Bevölkerungszahl eines Wahlkreises für die Bundestagswahl maximal um 15 Prozent von der durchschnittlichen Bevölkerungszahl der Wahlkreise abweichen[1]. Es sei jedoch erwähnt, dass es durchaus auch Anwendungsbeispiele gibt, bei denen es ein großes Ungleichgewicht zwischen der Größe zweier Wählergruppen gibt. So etwa, wenn eine der Gruppen aus einem Expertenkomittee zusammengesetzt ist, welches naturgemäß deutlich weniger Mitglieder enthält.
Die von Bartholdi et al. und Hemaspaandra et al. definierten Probleme wurden bereits für eine Vielzahl von Wahlregeln untersucht. Zu nennen sind dabei unter anderem die Arbeiten von Erdélyi et al. [24], Erdélyi et al. [26], Faliszewski et al. [29], Neveling und Rothe [61, 62], Menton [56] sowie Menton und Singh [57].
Darüber hinaus wurden in einer Vielzahl von Arbeiten verwandte Modelle definiert und für diverse Wahlregeln analysiert. So definierten Erdélyi et al. [25] drei neuen Probleme zur Partitionierung der Wählermenge. Diese sind von der Modellierung her sehr ähnlich zu der von Bartholdi et al. definierten Version, versuchen jedoch durch unterschiedliche Einschränkungen bei der Partitionierung reale Gegebenheiten zu modellieren. Ein weiteres Problem zeichnet sich dadurch aus, dass die Menge der Wähler auch in mehr als zwei Gruppen aufgeteilt werden kann. Schließlich wird noch das Problem definiert, dass nur ganze Wählergemeinschaften in eine von zwei Gruppen sortiert werden können. In den Arbeiten von Borodin et al. [7], Cohen-Zemach et al. [15] und Ito et al. [42] werden Graphen für eine geografische Einordnung der Wähler genutzt, sodass die beim Gerrymandering üblichen örtlichen Einschränkungen bei der Partitionierung der Wählermenge berücksichtigt werden können. Ein anderer Ansatz, um örtliche Gegebenheiten einzubinden, wurde von Lewenberg et al. [46]

1 Nach § 3 Abs. 1 Satz 1 Nr. 2, 3 und 5 Bundeswahlgesetz (BWG).

verfolgt. Dabei betten sie eine Wahl in den $\mathbb{R}^2$ ein und lassen eine Einteilung in verschiedene Wahldistrikte nur indirekt durch die Platzierung von Wahllokalen zu. Ein Wähler muss dann am ihm nächstgelegenen Wahllokal abstimmen. Die Platzierung von Wahllokalen hat noch weitere Auswirkungen auf die Wahl. Umso weiter weg oder umso schwieriger das nächste Wahllokal zu erreichen ist, umso eher ist davon auszugehen, dass einige Wähler ihr Wahlrecht nicht in Anspruch nehmen. Genau dieser Umstand wird in der Arbeit von Fitzsimmons und Lev [32] untersucht.

4.3 BESTECHUNG VON WAHLEN

In Kapitel 7 wird die Bestechung von Wahlen untersucht. Neben der Manipulation und der Kontrolle von Wahlen ist Bestechung eine weitere unerwünschte Form der Einflussnahme auf eine Wahl. Während allerdings die Manipulation und die Kontrolle von Wahlen als Probleme bereits Ende der 80er Jahre bzw. Anfang der 90er Jahre definiert und untersucht wurden, wurde die Bestechung erst Mitte der 2000er Jahre von Faliszewski et al. [27] definiert. Das erste dort definierte Bestechungsproblem ist ε-Bribery und wurde für eine Wahlregel ε wie folgt definiert.

ε-Bribery	
Gegeben:	Eine Wahl (C, V), ein ausgewählter Kandidat $p \in C$ und eine natürliche Zahl k.
Frage:	Ist es möglich durch Bestechung von höchstens k Wählern p zum Gewinner der Wahl zu machen?

Wird ein Wähler bestochen, so kann dessen Stimme auf beliebige Weise verändert werden. Damit stellt ε-Bribery eine gewisse Verallgemeinerung des Manipulationsproblems ε-Manipulation dar. In der selben Arbeit definieren die Autoren das Problem ε-$Bribery, welches ε-Bribery noch weiter verallgemeinert. Hier wird angenommen, dass jeder Wähler einen individuellen Preis für die Änderung seiner Stimme verlangt und einem Angreifer zusätzlich ein Budget zum Bestechen zur Verfügung steht.
Diese Probleme wurden seitdem in vielen weiteren Arbeiten für verschiedene Wahlregeln untersucht, wie etwa in den Arbeiten von Xia [74], Lin [47], Brelsford et al. [11], Faliszewski et al. [31], Faliszewski et al. [28] und Reisch et al. [65].
Seit der Formulierung des ersten Bestechungsproblems hat sich die Bestechung genau wie die Manipulation und Kontrolle als eigenständiger Zweig innerhalb der Computational Social Choice entwickelt.
Zu der Untersuchung von verschiedenen Wahlregeln hinsichtlich der vorgestellten Bestechungsprobleme wurden eine Reihe weiterer interessanter Bestechungsprobleme definiert und untersucht.

Ein besonders interessantes Problem ist SWAP-BRIBERY, welches von Elkind et al. [23] definiert wurde und eine Verallgemeinerung des Problems MICROBRIBERY von Faliszewski et al. [29] darstellt. Hier kann die Stimme eines Wählers im Allgemeinen nicht durch eine feste Geldeinheit beliebig geändert werden. Stattdessen hat jeder Wähler eine Kostenfunktion, welche für jedes Paar von Kandidaten angibt wie viel eine Verstauchung dieses Paares (auch Swap genannt) kostet. Die Gesamtkosten für die Transformation der ehrlichen Stimme eines Wählers in die veränderte Stimme ergeben sich dann aus der Summe der Kosten der Swaps, welche für diese Transformation benötigt werden. Eine positivere Interpretation ergibt sich, wenn man das SWAP-BRIBERY Problem, wie weiter oben erwähnt, als Messinstrument für das Abschneiden einzelner Kandidaten bei einer Wahl verwendet. Die Anzahl der Swaps, die ein Kandidat benötigt um eine gegebene Wahl zu gewinnen, kann als Maß genutzt werden um zu bestimmen wie knapp dieser Kandidat vom Sieg entfernt ist. Besonders interessant ist diese Art des Vergleichs für Wahlregeln, bei welchen den Kandidaten am Ende der Wahl kein Punktestand zugeordnet werden kann, wie etwa bei rundenbasierten Wahlregeln.
Ein Spezialfall von SWAP-BRIBERY ist SHIFT-BRIBERY, welches ebenfalls in der Arbeit von Elkind et al. [23] definiert wurde. Hier darf sich in einer Stimme nur die Position vom ausgewählten Kandidaten relativ zu der Position der anderen Kandidaten verändern. Auch dieses Problem lässt eine positive Betrachtungsweise zu. So kann ein Wahlteam eine vollzogene Wahl analysieren und planen, wie sie in der nächsten Wahl werben müssen oder welche Versprechen an die Wähler gegeben werden müssen um künftig besser abzuschneiden. Ähnlich wie bei den zuvor diskutierten Kontrollproblemen lassen sich auch Bestechungsprobleme in einer destruktiven Variante definieren, in der es darum geht einen ausgewählten Kandidaten am Sieg der Wahl zu hindern. Die destruktive Version von SHIFT-BRIBERY wurde zuerst von Kaczmarczyk und Faliszewski [43] definiert. In Kapitel 7 widmen wir uns diesem Problem sowohl in der konstruktiven als auch in der destruktiven Version für verschiedene iterative Wahlregeln.

Das Problem SHIFT-BRIBERY wurde für viele Wahlregeln hinsichtlich der klassischen und parametrisierten Komplexitätstheorie sowie mittels Approximationsalgorithmen untersucht. Zu nennen sind dabei unter anderem die Arbeiten von Boehmer et al. [6], Schlotter et al. [70], Elkind und Faliszewski [21], Faliszewski et al. [30], sowie von Bredereck et al. [9]. Eine weitere Richtung bei der Untersuchung von SHIFT-BRIBERY haben Elkind et al. [22] eingeschlagen, indem sie das Problem im Hinblick auf Stimmen untersucht haben, die als *single-peaked* oder *single-crossing* bezeichnet werden, worauf wir hier jedoch nicht im Detail eingehen werden.
Nah verwandt zum SHIFT-BRIBERY Problem ist das COMBINATORIAL SHIFT-BRIBERY Problem, welches Bredereck et al. [10] definierten. Ge-

nau wie beim SHIFT-BRIBERY Problem darf sich hier nur die Position des ausgewählten Kandidaten relativ zu der Position der anderen Kandidaten ändern. Um den Einfluss von Werbung zu modellieren, werden hier jedoch hauptsächlich Eingriffe betrachtet durch die mehrere Wähler gleichzeitig beeinflusst werden. Eine sicherlich nicht ganz unproblematische Annahme, welche wir bei den bisher vorgestellten Bestechungsvarianten getätigt haben, ist, dass jeder Wähler für eine bestimmte Summe bereit ist, den ausgewählten Kandidaten an eine beliebige Position seiner Stimme zu setzen. Um diese Annahme abzuschwächen, definieren Dey et al. [18] das Problem FRUGAL BRIBERY. Hier ist die Bestechung nur möglich, wenn der ausgewählte Kandidat in der Gunst des zu bestechenden Wählers ohnehin weiter vorne steht als der zu erwartende Gewinner der Wahl. Einer ähnlichen Logik folgen Yang et al. [75] bei ihren Untersuchungen zum BRIBERY Problem, indem sie die zusätzliche Einschränkung einführen, dass sich die veränderte Stimme nicht zu sehr von der echten Stimme unterscheiden darf. Schließlich sei noch die Arbeit von Chen et al. [14] erwähnt, in welcher eine BRIBERY-Instanz mit einem Budget für einen Verteidiger ausgestattet wird. Dieser soll die Wähler durch eine Zahlung dazu motivieren sich gegen etwaige Bestechungsversuche zu widersetzen und bei ihrer ehrlichen Stimme zu bleiben. In gewisser Weise findet sich dieser Ansatz bei der Entlohnung diverser Mandatsträger wieder. Der Kritik, dass diese oft zu üppig sei, wird gerne mit dem Argument begegnet, dass die Entscheidungen der Mandatsträger so vor äußeren Einflüssen geschützt werden soll.

5 KONTROLLE BEI VETO- UND PLURALITY-WAHLEN

In diesem Kapitel wird die Komplexität verschiedener Kontrollprobleme für Veto- und Plurality-Wahlen untersucht. Die Arbeit wurde in der Fachzeitschrift [53] veröffentlicht:

C. Maushagen und J. Rothe. „Complexity of Control by Partitioning Veto Elections and of Control by Adding Candidates to Plurality Elections". In: *Annals of Mathematics and Artificial Intelligence* 82.4 (2018), S. 219–244.

Die Arbeit umfasst und erweitert Resultate aus der bei ECAI 2016 veröffentlichten Arbeit [50] und der bei AAMAS 2017 veröffentlichten Arbeit [52].

5.1 ZUSAMMENFASSUNG

In dieser Arbeit untersuchen wir die Wahlregeln Veto und Plurality hinsichtlich verschiedener Kontrollprobleme, welche von Bartholdi et al. [5] und Hemaspaandra et al. [38] eingeführt wurden.
Im ersten Teil der Arbeit betrachten wir für die Wahlregel Veto die Kontrollprobleme zur Partitionierung der Kandidaten und Partitionierung der Wähler, welche im Abschnitt 4.2 definiert wurden. Es zeigt sich, dass die Kontrollprobleme zur Partitionierung der Wählermenge effizient lösbar sind, wenn als Gleichstandsbrechungsregel TE genutzt wird. Dieses Ergebnis gilt dabei sowohl für den konstruktiven als auch für den destruktiven Fall, sowie für das unique-winner Modell und das nonunique-winner Modell. Alle anderen untersuchten Kontrollprobleme bezüglich der Partitionierung der Wähler- oder Kandidatenmenge sind ebenfalls NP-vollständig.
Im zweiten Teil der Arbeit wenden wir uns der Wahlregel Plurality zu. Hier betrachten wir das Kontrollproblem des Hinzufügens von Kandidaten. Dieses ist wie folgt definiert.

ε-Constructive-Control-by-Adding-Candidates	
Gegeben:	Eine Menge C und eine Menge D von Kandidaten, $C \cap D = \emptyset$, eine Liste V von Wählern über $C \cup D$, ein ausgezeichneter Kandidat $p \in C$ und eine natürliche Zahl $k \leq \lvert D \rvert$.
Frage:	Gibt es eine Teilmenge $D' \subset D$ mit $\lvert D' \rvert \leq k$, sodass p unter dem Wahlsystem ε die Wahl $(C \cup D', V)$ gewinnt?

Dieses Problem wurde bereits zuvor in der Arbeit von Chen et al. [12] behandelt, welche das Problem hinsichtlich seiner parametrisier-

ten Komplexität analysieren. Dabei haben die Autoren untersucht, welchen Einfluss das Festhalten des Parameters *Anzahl der Wähler* auf die Komplexität verschiedener Kontrollprobleme hat. Der dort angegebene Beweis für die W[1]-Härte für Plurality ist fehlerhaft. Dieser Umstand wird in unserer Arbeit aufgezeigt und analysiert. Zudem konnten wir durch eine Änderung der dort angegebenen Konstruktion einen vollständigen Beweis der W[1]-Härte angeben.

5.2 EIGENER ANTEIL

Die Arbeit wurde zusammen mit Jörg Rothe geschrieben. Die technischen Teile der Arbeit sind mir zuzuschreiben.

Ann Math Artif Intell (2018) 82:219–244
https://doi.org/10.1007/s10472-017-9565-7

Complexity of control by partitioning veto elections and of control by adding candidates to plurality elections

Cynthia Maushagen[1] · **Jörg Rothe**[1]

Published online: 26 October 2017

Abstract Control by partition refers to situations where an election chair seeks to influence the outcome of an election by partitioning either the candidates or the voters into two groups, thus creating two first-round subelections that determine who will take part in a final round. The model of partition-of-voters control attacks is remotely related to "gerrymandering" (maliciously resizing election districts). While the complexity of control by partition has been studied thoroughly for many voting systems, there are no such results known for the important veto voting system. We settle the complexity of control by partition for veto in a broad variety of models. In addition, by giving a counterexample we observe that a reduction from the literature (Chen et al. 2015) showing the parameterized complexity of control by adding candidates to plurality elections, parameterized by the number of voters, is technically flawed, and we show how this reduction can be adapted to make it correct.

Keywords Computational social choice · Voting · Veto election · Control complexity

Mathematics Subject Classification (2010) 91B14 · 68Q17 · 68Q15 · 68T99

Preliminary versions of this paper have been presented at the 14th International Symposium on Artificial Intelligence and Mathematics (ISAIM 2016) and have appeared in the proceedings of the 22nd European Conference on Artificial Intelligence (ECAI 2016) [33] and of the 16th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2017) [34]. This paper combines some of their results, unifies and simplifies their proofs, and adds discussion and examples.

✉ Cynthia Maushagen
maushagen@cs.uni-duesseldorf.de

Jörg Rothe
rothe@cs.uni-duesseldorf.de

[1] Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, Universitätsstraße 1, Düsseldorf, Germany

1 Introduction

Along with manipulation [1, 9] and bribery [15, 18], electoral control [2, 24] has been the focus of much attention in computational social choice; see the book chapters by Faliszewski and Rothe [19] and Baumeister and Rothe [5] for a survey of the related results. Control scenarios model settings where an external agent, commonly referred to as the *chair*, seeks to influence the outcome of an election by such actions as adding, deleting, or partitioning either the candidates or the voters. The above-mentioned chapters and the papers cited therein comprehensively describe applications of voting in artificial intelligence, multiagent systems, ranking algorithms, meta-websearch, etc., and they discuss how computational complexity can be used to provide some protection against manipulation, bribery, and control attacks. In particular, they give real-world examples (see also Section 2.2) of the various control types introduced by Bartholdi et al. [2] for the constructive control goal where the chair aims at making a given candidate win and by Hemaspaandra et al. [24] for destructive control where the chair's goal is to prevent a given candidate's victory.

We here focus mostly on control-by-partition scenarios for veto elections (in Sections 3 and 4); in addition, we will be concerned with constructive control by adding candidates for plurality elections in Section 5. Plurality and veto belong to the important class of scoring protocols and are two of the most simple and most important, widely used voting systems. The voters in plurality elections give one point to their most favorite candidate and the voters in veto elections give one point to each candidate except their most despised candidate, and whoever scores the most points in either plurality or veto wins.

Related work The complexity of control has been studied for many voting systems. Control for plurality, Condorcet, and approval voting has first been studied by Bartholdi et al. [2] in the constructive variant and by Hemaspaandra et al. [24] in the destructive variant (see also the related work by Betzler and Uhlmann [7]). Later on, control results have been obtained by Faliszewski et al. [18] for Copeland; by Erdélyi et al. [12] for Bucklin and fallback voting; by Parkes and Xia [40] for Schulze voting; by Russel [43], Elkind et al. [11], Loreggia et al. [32], Chen et al. [8], and Neveling and Rothe [36, 37] for Borda elections; by Erdélyi et al. [14] for a certain variant of approval voting (namely, sincere-strategy preference-based approval voting); and by Menton [35] for (normalized) range voting.

Hemaspaandra et al. [25] proposed a way to broaden complexity-theoretic resistance to control via hybrid elections. A dichotomy result for constructive control by adding voters is due to Hemaspaandra et al. [28] and another dichotomy result is due to Hemaspaandra and Schnoor [29]. Faliszewski et al. [16] introduced and studied multimode control attacks on elections that combine various standard control scenarios. Another paper by Faliszewski et al. [17] is concerned with the complexity of control in weighted elections. A study of online voter and candidate control in sequential elections is due to Hemaspaandra et al. [26, 27]. More results on electoral control (and bribery) can be found in recent book chapters [5, 19].

The veto rule has been studied, for example, by Conitzer et al. [9] with respect to coalitional weighted manipulation in terms of classical complexity, by Walsh [44] with respect to the phase transition when manipulating veto empirically (i.e., in terms of typical-case complexity), by Faliszewski et al. [15] with respect to bribery, and by Lin [31] and Chen et al. [8] with respect to control by adding or deleting candidates or voters in terms of both classical and parameterized complexity. Perhaps a bit surprisingly, complexity results about control by partition of either candidates or voters for the important veto voting system are missing to date. This is all the more surprising as control by partition of voters provides a simplified model of gerrymandering (i.e., maliciously resizing election districts), a particularly natural

control type known from the real world. One reason why these control scenarios have been neglected so far for veto may be that proofs for control by partition tend to be technical and challenging.

Our contribution We settle all cases of control by partition of either candidates or voters for veto elections in terms of their classical complexity. In particular, this refers to constructive and destructive control by partition of either candidates or voters in both the ties-eliminate and the ties-promote model and in both the unique-winner and the nonunique-winner model (see Section 2.2 for definitions). For control by partition of candidates, we further distinguish between the cases with and without run-off. Table 1 gives an overview of our results on the computational complexity of control by partition for veto elections.

Since most of the known results on control by partition are in the original model as suggested by Bartholdi et al. [2] where the candidates or voters can be partitioned into two sets of arbitrary sizes, we will focus on this model too, in order to allow for comparability of results. However, we suggest to also study these problems for veto in the more refined models due to Erdélyi et al. [13] that restrict such partitions to sets of roughly the same size and due to Puppe and Tasnádi [41] that take geographical constraints into account when resizing election districts.

In addition, we observe that a reduction due to Chen et al. [8, Theorem 1] showing the parameterized complexity of constructive control by adding candidates to plurality elections, where the parameter is the number of voters, is technically flawed. Specifically, we give a counterexample showing that their reduction maps a no-instance of the problem MULTI-COLORED-CLIQUE to a yes-instance of this control problem, and we show how this reduction can be adapted so as to make it correct.

Organization This paper is organized as follows. In Section 2, we define the considered voting systems and control problems and briefly present the needed notions from complexity theory. We then study for veto elections the complexity of control by partition of voters in Section 3 and that of control by partition of candidates in Section 4. In Section 5, we observe the above-mentioned technical flaw in a reduction due to Chen et al. [8] by providing a counterexample, and we show how to fix this reduction. Finally, we conclude in Section 6 and briefly discuss some lines of future research.

2 Preliminaries

In this section, we define the voting systems we will consider, plurality and veto, and the control-by-partition problems we will study for veto. We will also show that veto voting is

Table 1 Overview of complexity results for control by partition in veto elections. Control types are denoted as is standard [5, 19]; they are defined in Section 2.2. "R" means that veto is resistant to this type of control and "V" means it is vulnerable to this type. Results established in this paper: Theorem 1 (marked by ❄), Theorem 2 (♡), Theorem 3 (♠), Corollary 1 (◇), Theorem 4 (†), Corollary 2 (‡), Theorem 5 (♮), Corollary 3 (♣), Theorem 6 (¶), and Corollary 4 (§)

control type	CPC-TE		CPC-TP		CRPC-TE		CRPC-TP		CPV-TE		CPV-TP	
	C	D	C	D	C	D	C	;D	C	D	C	D
unique-winner model	R$^{♮}$	R¶	R$^{♮}$	R§	R$^{♮}$	R¶	R$^{♮}$	R$^{♣}$	V$^{❄}$	V$^{♡}$	R$^{♠}$	R‡
nonunique-winner model	R¶	R¶	R¶	R$^{♣}$	R$^{♮}$	R¶	R$^{♮}$	R$^{♣}$	V$^{❄}$	V$^{♡}$	R†	R$^{◇}$

susceptible to all control scenarios considered here, and we will give the needed background on computational complexity.

2.1 Plurality and veto elections

An election is given by a pair (C, V), where C is a set of candidates and V a list of the voters' preferences over the candidates (which we will simply refer to as their votes). We will consider only preferences that are linear orders (strict rankings) with the left-most candidate being the most preferred one. For example, a vote $d\ c\ a\ b$ means that this voter prefers d to c, c to a, and a to b.

We will consider two well-known voting systems, plurality and veto. Plurality is the perhaps simplest and still very prominent positional scoring protocol, a class of important voting systems that are based on the candidates' positional scores. Besides plurality, this class contains, for example, the popular voting systems veto (defined below) and Borda count. Plurality and veto are defined as follows:

- In *plurality*, every voter gives one point to her most preferred candidate, and whoever scores the most points wins.
- In *veto*, every voter vetoes her least preferred candidate, which means that this candidate gets no point while all other candidates receive one point from this voter, and whoever scores the most points wins.

2.2 Control-of-partition problems

We consider control by partition of either candidates or voters, as defined by Bartholdi et al. [2] for constructive control and by Hemaspaandra et al. [24] for destructive control.[1] The problems defined below have been studied in many papers for a variety of voting systems; we refer to the book chapters by Faliszewski and Rothe [19] and Baumeister and Rothe [5] for an overview. Below, we will provide formal definitions of all problems studied here and will present real-world examples motivating the control scenarios we are interested in. In each such control-of-partition scenario, starting from a given election (C, V) and a distinguished candidate $c \in C$, we form two subelections—either (C_1, V) and (C_2, V) where C is partitioned into C_1 and C_2 (i.e., $C_1 \cap C_2 = \emptyset$ and $C_1 \cup C_2 = C$),[2] or (C, V_1) and (C, V_2) where V is partitioned into V_1 and V_2 (i.e., $V_1 \cap V_2 = \emptyset$ and $V_1 \cup V_2 = V$)—whose winners move forward to a final round if they survive the given tie-handling rule: either *ties-eliminate* (TE) that requires that only unique winners of a first-round subelection move forward, or *ties-promote* (TP) that requires that all winners of a first-round subelection move forward.

Such a partition of either C or V is the chair's control action, and the chair's goal is either to ensure that the distinguished candidate c wins the final round (in the *constructive* case) or to prevent c's victory (in the *destructive* case), where the final round is always held with all votes from V (restricted, of course, to the candidates taking part in the final round;

[1] Constructive control by adding candidates, also due to Bartholdi et al. [2], will be defined in Section 5 because this control type will be considered only there.

[2] Note that when we consider a subelection such as (C_1, V) with a subset $C_1 \subseteq C$ of the candidates, all votes in this subelection are restricted to C_1. For example, if the original election (C, V) with four candidates, $C = \{a, b, c, d\}$, contains a vote $d\ c\ a\ b$ and we consider a subelection (C_1, V) with two candidates, $C_1 = \{a, d\}$, then this vote is restricted to these two candidates: $d\ a$.

see Footnote 2). In the case of candidate control, we further distinguish between *run-off partition of candidates*, where the winners of (C_1, V) and (C_2, V) surviving the tie-handling rule face each other in the final run-off, and *partition of candidates*, where the winners of (C_1, V) surviving the tie-handling rule face all candidates of C_2 in the final round.

Let us briefly recall examples from the literature (in particular, from the book chapter by Baumeister et al. [3]) showing that control by partition does occur in the real world. Indeed, *control by run-off partition of candidates* models scenarios where a committee (comprising the voters) divides the alternatives (or candidates) into two groups, votes on each group separately, and then takes a final vote on the winners of both subelections. For example, when a computer science department is recruiting a new professor, the chair of the recruiting committee might partition the candidates into two groups (e.g., one for applicants from theoretical computer science and the other for applicants from practical computer science, or one group with established researchers and the other group with early-stage carreer researchers, or one with female and one with male researchers, or using any other criterion she might see fit her plans). *Control by (non-run-off) partition of candidates*, on the other hand, models scenarios where one group of alternatives (or candidates) is exempted from a "qualifying" stage,[3] while the others have to survive a qualification first, determining some winner(s) among them to participate in the final round. Finally, *control by partition of voters* models a primary system where the electorate is divided into two groups, each group voting over all candidates, and the subelection winners moving forward to a final round where everyone votes on them. Many important real-world scenarios correspond to this control type. The chair of the recruiting committee mentioned above, for example, might decide to hold two first-round subelections, one with all theoreticians of the department and one with all its practical computer scientists voting on the candidates, and then the winners of both move forward to a final round where everyone votes again. Or, maliciously resizing election districts (a behavior known as *gerrymandering*) is a particularly natural type of control occurring in the real world and modeled, in a simplified way, by partition of voters.

For each such control scenario, we can define a decision problem. As an example, we formally define the decision problem associated with constructive control by run-off partition of candidates in model TE for some given voting system $\mathscr{E}$:

$\mathscr{E}$-Constructive-Control-by-Run-off-Partition-of-Candidates-TE	
Given:	An election (C, V) and a distinguished candidate $c \in C$.
Question:	Can C be partitioned into C_1 and C_2 such that c is the unique $\mathscr{E}$ winner of the two-round election where the $\mathscr{E}$ winners of the two first-round subelections, (C_1, V) and (C_2, V), who survive tie-handling rule TE (i.e., only unique winners move forward) run against each other in a final round (with the votes from V correspondingly restricted in each subelection; see Footnote 2)?

The above problem (denoted by $\mathscr{E}$-CCRPC-TE) is defined in the *unique-winner model*. We will also consider the *nonunique-winner model* where the question is changed to ask whether c is a winner (possibly among several winners) of the final round,

[3]This often happens, for example, in tournaments where the home country and perhaps some other alternatives are automatically set as participants for the final round. Even though the winners of sports tournaments in general are not determined by voting (except, for example, in dancing contests or similar events), such a procedure is also well suited in the context of voting.

and we will always specify the winner model we are referring to. $\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-RUN-OFF-PARTITION-OF-CANDIDATES-TP (denoted by $\mathscr{E}$-CCRPC-TP) is defined analogously, except with TP used to determine who will move forward to the final run-off from the two first-round subelections (namely, all their winners). $\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-CANDIDATES-TE ($\mathscr{E}$-CCPC-TE) and $\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-CANDIDATES-TP ($\mathscr{E}$-CCPC-TP) are defined analogously to the above two problems, except that we now don't have a run-off among the winners (surviving the tie-handling rule) of two first-round subelections but instead the winner (in TE, provided there exists one) or all winners (in TP) of one first-round subelection, (C_1, V), run(s) against all candidates of C_2 in the final round. The four constructive control problems defined so far have a destructive variant each: $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-RUN-OFF-PARTITION-OF-CANDIDATES-TE ($\mathscr{E}$-DCRPC-TE), $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-RUN-OFF-PARTITION-OF-CANDIDATES-TP ($\mathscr{E}$-DCRPC-TP), $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-PARTITION-OF-CANDIDATES-TE ($\mathscr{E}$-DCPC-TE), and $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-PARTITION-OF-CANDIDATES-TP ($\mathscr{E}$-DCPC-TP). The difference is that the chair's destructive control goal is satisfied only if the distinguished candidate is not a unique winner (in the *unique-winner model*—note that this candidate might win, among other winners, and the destructive goal would still be satisfied in this winner model) or is not even a winner (in the *nonunique-winner model*).[4]

All eight control-by-partition-of-candidates problems above come in two winner models, the unique-winner and the nonunique-winner model, yielding 16 problems in total. However, Hemaspaandra et al. [23, Thm. 8 on p. 386] have shown that, for all voting systems, DCRPC-TP and DCPC-TP are in fact identical problems in the nonunique-winner model and DCRPC-TE and DCPC-TE are in fact identical problems in both winner models. This reduces the number of problems of control by partition of candidates from 16 to 13.

As to control by partition of voters,[5] there are eight decision problems for some given voting system $\mathscr{E}$: the two constructive problems $\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TE ($\mathscr{E}$-CCPV-TE) and $\mathscr{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TP ($\mathscr{E}$-CCPV-TP), their destructive counterparts $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TE ($\mathscr{E}$-DCPV-TE) and $\mathscr{E}$-DESTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TP ($\mathscr{E}$-DCPV-TP), and each of these four problems again coming in both winner models. The difference to the control-by-partition-of-candidates problems is merely that now the voters, V, not the candidates are partitioned into two groups, V_1 and V_2, yielding the first-round subelections (C, V_1) and (C, V_2) whose winners (that survive the tie-handling rule) move forward to the final round.

[4]Note that the TE rule naturally fits the unique-winner model, whereas the TP rule naturally fits the nonunique-winner model. However, for completeness, we will consider all combinations of winner model and tie-handling rule.

[5]Note that for control by partition of voters, a distinction between *with* and *without run-off* (as in control by partition of candidates) does not make any sense: Partition of voters *without run-off* would mean that the winners of the first-round subelection (C, V_1) surviving the tie-handling rule face all candidates in the final round, so the final round would always be just the original election (C, V). Therefore, only control by partition of voters *with run-off* is considered and, as is common in the literature, we omit mentioning explicitly "*with run-off*" and speak simply of "control by partition of voters." We also omit "RUN-OFF-" in the problem name and the "R" in its shorthand, as these are the default notations in the literature.

2.3 Immunity, susceptibility, vulnerability, and resistance

For a control type $\mathfrak{C}$ (such as constructive control by partition of voters in model TE), an election system $\mathscr{E}$ is said to be *immune to* $\mathfrak{C}$ if it is impossible for the chair to reach her control goal (e.g., to make the given candidate c a unique winner in the constructive case for the unique-winner model, or to ensure that c is not even a winner in the destructive case for the nonunique-winner model) via exerting control of type $\mathfrak{C}$; otherwise, $\mathscr{E}$ is said to be *susceptible to* $\mathfrak{C}$. In Example 1 below, we will show that the voting system we study here, veto, is susceptible to every type of control we have defined above (in both winner models).

Example 1 Let $C = \{a, b, c, d\}$ be a set of candidates and V the following list of votes:

$$v_1 = a\ b\ c\ d, \quad v_2 = c\ d\ a\ b, \quad v_3 = c\ d\ a\ b, \quad v_4 = b\ c\ a\ d,$$
$$v_5 = b\ c\ d\ a, \quad v_6 = b\ d\ a\ c, \quad v_7 = d\ b\ c\ a.$$

In election (C, V), candidate c is the only veto winner. Partition V into $V_1 = \{v_1, v_5, v_6\}$ and $V_2 = \{v_2, v_3, v_4, v_7\}$. The only veto winner of subelection (C, V_1) is b and the only veto winner of subelection (C, V_2) is c, so only b and c move forward to the final round, regardless of the tie-handling rule. Candidate b wins the final round with three points more than c. Since b can be made even a unique winner and since it can be ensured that c is not even a winner by this control action, it follows that veto is susceptible to both constructive and destructive control by partition of voters in models TE and TP, in both winner models.

Now consider election (C, V') with $V' = \{v_1, v_2, v_3, v_4, v_5\}$. Candidate c is the only veto winner of this election as well, since c does not receive any veto while all the remaining candidates get at least one veto. If we partition C into $C_1 = \{b, c\}$ and $C_2 = \{a, d\}$ then b wins subelection (C_1, V) with one point more than c. In subelection (C_2, V), candidate d is the only veto winner, so the final run-off is with b and d, and in direct comparision b beats d. Since again b can be made even a unique winner and since it can again be ensured that c is not even a winner by this control action, it follows that veto is susceptible to both constructive and destructive control by run-off partition of candidates in models TE and TP, in both winner models. If for the same partition of candidates in (C, V') there is no second first-round subelection then both candidates of C_2 move forward to the final round, $(\{a, b, d\}, V')$, where a scores one point more than b and d and thus is the unique veto winner of the election. Thus, as above, veto is also susceptible to both constructive and destructive control by partition of candidates in models TE and TP, in both winner models.

If an election system $\mathscr{E}$ is susceptible to some control type $\mathfrak{C}$, it is common to study the computational complexity of the associated control problem: We say $\mathscr{E}$ is *vulnerable to* $\mathfrak{C}$ if the control problem corresponding to $\mathfrak{C}$ can be solved in polynomial time, and we say $\mathscr{E}$ is *resistant to* $\mathfrak{C}$ if $\mathfrak{C}$ is NP-hard.

2.4 Computational complexity

We assume that the reader is familiar with the basic notions of computational complexity, such as the complexity classes P (deterministic polynomial time) and NP (nondeterministic polynomial time) and with the notions of NP-hardness and NP-completeness, based on polynomial-time many-one reducibility. For more background, we refer to the books by Garey and Johnson [20], Papadimitriou [39], and Rothe [42].

In Section 5, we will also be concerned with *parameterized* complexity. In particular, we consider a result about W[1]-hardness. W[1] is a parameterized complexity class that in some sense corresponds to the classical complexity class NP, and just as NP-hardness indicates that a problem is infeasible to solve in the sense of classical complexity theory (i.e., has no polynomial-time algorithm unless P = NP), W[1]-hardness can be taken as strong evidence that a problem is not even fixed-parameter tractable. For more background on parameterized complexity and fixed-parameter tractability, we refer to the books by Downey and Fellows [10] and Niedermeier [38].

3 Control by partition of voters in veto elections

In this section, we present our results for controlling veto elections by partiton of voters in both tie-handling models, TE and TP, and in both the unique-winner and the nonunique-winner model.

3.1 Veto-CCPV-TE and Veto-DCPV-TE

We show that veto is vulnerable to constructive and destructive control by partition of voters in model TE for both winner models, and we start with the constructive case. Essentially, the polynomial-time algorithm used to prove Theorem 1 exploits the fact that, due to the TE model, control is impossible only if either there are two candidates and the distinguished candidate is not already a veto winner (in the unique-winner model: is not already the only veto winner) of the given election, or there are more than two candidates and some candidate other than the distinguished candidate is not vetoed by any voter. In all other cases, it is easy to find a successful partition that ensures the distinguished candidate's victory.

Theorem 1 *Veto*-CCPV-TE *is in* P *in both the unique-winner and the nonunique-winner model.*

Proof The following polynomial-time algorithm solves the problem. Given an election (C, V) with n votes in V and a candidate $c \in C$, it proceeds as follows:

1. If there are no more than two candidates, then if c already is a winner (in the unique-winner model: the only winner) of (C, V), control is possible via the trivial partition $(V, \emptyset)$, so accept; otherwise, control is impossible, so reject.
2. Otherwise (i.e., if $|C| > 2$), if $score(d) = n$ for some $d \in C \setminus \{c\}$, control is impossible, so reject.
3. Otherwise (i.e., if $|C| > 2$ and $score(d) < n$ for all $d \in C \setminus \{c\}$), it is safe to accept, since control is possible via the partition (V_1, V_2) of V that puts all voters who veto c into V_1 and all other voters into V_2.

The above algorithm runs in polynomial time and is correct. This is not hard to see for step 1. Further, it is impossible for c to defeat the candidate d with $score(d) = n$ in step 2 (as d scores the maximum number of points in each first-round subelection, no matter how V is partitioned, which makes it impossible for c to win alone in any subelection). And in step 3, no candidate from V_1 can move to the final round, because either V_1 is empty (in case no one vetoes c) or each of the at least two candidates other than c wins subelection (C, V_1) with the same score and, therefore, will be eliminated in model TE. On the other

hand, since $score(d) < n$ for all $d \in C \setminus \{c\}$, each candidate $d \neq c$ is vetoed by at least one voter ending up in V_2, whereas c is not vetoed by any voter in V_2 and thus wins subelection (C, V_2) and the final run-off. This argument applies to both the unique-winner and the nonunique-winner model. □

A similar algorithm works in the destructive case. Note that Theorem 2 follows immediately from Theorem 1 for the unique-winner model,[6] but not for the nonunique-winner model. Therefore, we present a proof (which in fact works for both winner models).

Theorem 2 *Veto*-DCPV-TE *is in* P *in both the unique-winner and the nonunique-winner model.*

Proof Given an election (C, V) and a distinguished candidate c, our algorithm works as follows:

1. If $|C| = 1$, control is impossible, so reject.
2. If $|C| = 2$, determine the veto winners of (C, V). Control is impossible in the nonunique-winner model if c wins (and in the unique-winner model if c wins alone), so reject in this case. Otherwise, control is possible via the trivial partition $(V, \emptyset)$, so accept.
3. If $|C| > 2$, it is safe to outright accept, since control is always possible: Fix some candidate $d \neq c$ and partition V into (V_1, V_2) such that V_1 contains all voters vetoing d and V_2 contains all remaining voters.

The above algorithm runs in polynomial time and its correctness is straightforward for steps 1 and 2, while it follows for step 3 from the observation that if either c or d is vetoed by everyone then (V_1, V_2) will be trivial (either $(\emptyset, V)$ or $(V, \emptyset)$) and will thus prevent c from winning, and if neither c nor d is vetoed by everyone then there is a candidate e, $c \neq e \neq d$, who ties for winner with c in (C, V_1), while d ties-or-defeats c in (C, V_2); in either case, c cannot move forward to the final round due to model TE. □

3.2 Veto-CCPV-TP and Veto-DCPV-TP

While veto is vulnerable to constructive and destructive control by partition of voters in model TE for both winner models, the corresponding decision problems turn out to be NP-complete if we change the tie-handling rule to TP. We first deal with the unique-winner model and will turn to the nonunique-winner model later on in Theorem 4.

Theorem 3 *Veto*-CCPV-TP *is* NP*-complete in the unique-winner model.*

Proof Membership of veto-CCPV-TP in NP is obvious. To show that it is NP-hard, we reduce from ONE-IN-THREE-POSITIVE-3SAT, an adaption from the well-known NP-complete problem ONE-IN-THREE-3SAT where the clauses of the given boolean formula do not contain any negated variables [20, p. 259]:

[6] As noted by Hemaspaandra et al. [24, Footnote 5 on p. 257], for voting systems that always have at least one winner (such as veto), any destructive control problem in the unique-winner model disjunctively truth-table reduces to the corresponding constructive control problem in the nonunique-winner model. Note that P is closed under this reducibility, i.e., each problem that disjunctively truth-table reduces to some P problem is itself in P.

ONE-IN-THREE-POSITIVE-3SAT

Given: A set X of boolean variables, a set S of clauses over X, each containing exactly three unnegated variables.

Question: Does there exist a truth assignment to the variables in X such that exactly one literal is set to true for each clause in S?

Let (X, S) be an instance of ONE-IN-THREE-POSITIVE-3SAT with $X = \{x_1, \ldots, x_m\}$ and $S = \{S_1, \ldots, S_n\}$, where we may assume, without loss of generality, that $n > 1$. Construct an election (C, V) with distinguished candidate $c \in C$ by defining $C = X \cup \{c, w\}$, where the elements of X from now on will also be viewed as candidates, and the list V of votes as follows:

number of votes	preference	
$2n^2 + 1$	$w\ c\ \cdots\ x_i$	for each $i \in \{1, \ldots, m\}$
$n - 1$	$w\ \cdots\ c$	
1	$c\ \cdots\ w\ S_j \setminus \{x_i\}$	for each $j \in \{1, \ldots, n\}$ and $x_i \in S_j$
$2n$	$w\ \cdots\ c\ S_j$	for each $j \in \{1, \ldots, n\}$

The reduction can be computed in polynomial time. The election contains $m + 2$ candidates and $(2n^2 + 1)m + n - 1 + 3n + 2n^2 = (2n^2 + 1)m + 2n^2 + 4n - 1$ votes, and w is the only winner of the election because the candidates have the following scores:

$$\begin{aligned} score(c) &= (2n^2 + 1)m + 3n + 2n^2, \\ score(w) &= (2n^2 + 1)m + n - 1 + 3n + 2n^2, \text{ and} \\ score(x_i) &\leq (2n^2 + 1)(m - 1) + n - 1 + 3n + 2n^2. \end{aligned}$$

We will now show that (X, S) is in ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, c) is in veto-CCPV-TP in the unique-winner model.

($\Rightarrow$) Let (X, S) be a yes-instance of ONE-IN-THREE-POSITIVE-3SAT. Then there is a subset $U = \{u_1, \ldots, u_\ell\}$ of X (renaming its elements for convenience) such that for each clause S_j we have $|U \cap S_j| = 1$. Partition V into V_1 and V_2 such that V_1 contains exactly one vote of the form $w\ c\ \cdots\ x_i$ for each candidate $x_i \in X \setminus U$ and V_2 contains all remaining votes. We claim that c will be made the unique winner by this partition.

In the first subelection, (C, V_1), none of the candidates c, w, and $u_i \in U$ is vetoed and thus they all have the maximum score, whereas all candidates $x_i \in X \setminus U$ score exactly one point fewer. Therefore, candidates c, w, and all $u_i \in U$ proceed to the final round from this subelection. In the second subelection, (C, V_2), only w does not receive any vetoes, which means that only w moves forward to the final round. Thus $(\{c, w\} \cup U, V)$ is the final-round election, and we have the following scores:

$$\begin{aligned} score(c) &= (2n^2 + 1)m + 3n + 2n^2, \\ score(w) &= (2n^2 + 1)m + n - 1 + 2n + 2n^2, \text{ and} \\ score(u_i) &\leq (2n^2 + 1)(m - 1) + n - 1 + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

Candidate w is vetoed in each vote in V that results from a vote of the form $c\ \cdots\ w\ S_j \setminus \{u_i\}$; these are n vetoes in total. At the same time, candidate c still receives only $n - 1$ vetoes from the votes of the form $w\ \cdots\ c$ and evades any further vetoes in the final round. To show that

c is the only winner, we have to show $score(c) > score(w)$ and $score(c) > score(u_i)$. The first inequality $score(c) > score(w)$ is equivalent to

$$(2n^2 + 1)m + 3n + 2n^2 > (2n^2 + 1)m + n - 1 + 2n + 2n^2,$$

which in turn is equivalent to $3n > 3n - 1$, which is true. The second inequality $score(c) > score(u_i)$ is equivalent to

$$(2n^2 + 1)m + 3n + 2n^2 > (2n^2 + 1)(m - 1) + n - 1 + 3n + 2n^2,$$

which in turn is equivalent to $2n^2 + 1 > n - 1$, which again is true. It follows that c alone scores the most points in the final-round election and thus is the unique winner, i.e., (C, V, c) is a yes-instance of veto-CCPV-TP in the unique-winner model.

($\Leftarrow$) Let (X, S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. Then, for any partition of X into X_1 and X_2, let k_i be the number of clauses containing i literals from X_1, $0 \leq i \leq 3$. We have $k_0 + k_1 + k_2 + k_3 = n$. Since we started from a no-instance of ONE-IN-THREE-POSITIVE-3SAT, we know that $k_1 \neq n$. We have to show that (C, V, c) is a no-instance of veto-CCPV-TP as well. Since w is vetoed by no voter in election (C, V), w will always (regardless of which voter partition is chosen) be a winner of both first-round subelections and will move to the final round. To show that it is impossible for c to become a unique winner of the final round for any partition of voters, we will show that w is always a winner in each possible final round. We consider two cases.

Case 1: Only the candidates c and w participate in the final round, leading to a direct comparison between them. Only $3n$ voters prefer c to w while all the remaining $(2n^2+1)m+2n^2+n-1$ voters prefer w to c. It follows that c is not a winner of the election.

Case 2: The candidates c, w and some candidates from X (say, $U = \{u_1, \ldots, u_\ell\}$ with $\emptyset \neq U \subseteq X$, renaming U's elements for convenience) participate in the final round. We consider three subcases that differ in terms of the number of clauses S_j for which $S_j \cap U = \emptyset$.

Case 2.1: $k_0 = 1$. Then we have the following bounds on the scores of the candidates in the final-round election:

$$\begin{aligned} score(c) &= (2n^2 + 1)m + 3n + (n - 1)2n, \\ score(w) &\geq (2n^2 + 1)m + n - 1 + 2(n - 1) + 2n^2, \text{ and} \\ score(u_i) &\leq (2n^2 + 1)(m - 1) + n - 1 + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

The score of w is at least

$$(2n^2 + 1)m + n - 1 + 2(n - 1) + 2n^2 - ((2n^2 + 1)m + 3n + (n - 1)2n) = 2n - 3$$

higher than the score of c, which for $n > 1$ is positive. Furthermore, the score of w is at least

$$(2n^2+1)m+n-1+2(n-1)+2n^2-((2n^2+1)(m-1)+n-1+3n+2n^2) = 2n^2-n-1$$

higher than the score of any u_i, $1 \leq i \leq \ell$, which for $n > 1$ is positive. It follows that w is the unique winner of the final round.

Case 2.2: $k_0 \geq 2$. Then we have the following bounds on the scores of the candidates in the final round:

$$\begin{aligned} score(c) &= (2n^2+1)m + 3n + (n-k_0)2n, \\ score(w) &\geq (2n^2+1)m + n - 1 + 2n^2, \text{ and} \\ score(u_i) &\leq (2n^2+1)(m-1) + n - 1 + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

For $k_0 \geq 2$ and $n > 1$, we have $2n + 1 < 2k_0n$, which implies

$$score(c) = (2n^2+1)m + 3n + (n-k_0)2n < (2n^2+1)m + n - 1 + 2n^2 \leq score(w).$$

Furthermore, since $2n^2 + 1 > 3n$ for $n > 1$, we also have $score(u_i) < score(w)$ for each i, $1 \leq i \leq \ell$, so w is the unique winner of the final round.

Case 2.3: $k_0 = 0$. That is, there does not exist any clause S_j such that $S_j \cap U = \emptyset$. Since we have started from a no-instance (X, S) of ONE-IN-THREE-POSITIVE-3SAT, there is at least one clause S_j with $|S_j \cap U| \geq 2$. Thus we have the following bounds on the scores of the candidates in the final round:

$$\begin{aligned} score(c) &= (2n^2+1)m + 3n + 2n^2, \\ score(w) &\geq (2n^2+1)m + n - 1 + 2n + 1 + 2n^2, \text{ and} \\ score(u_i) &\leq (2n^2+1)(m-1) + n - 1 + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

Candidate w gets at least $2n + 1$ points from the voters of the third group, since at least one clause contains two candidates u_i, so w does not receive any vetoes from the three voters corresponding to this clause, and w receives at most one veto for every other clause. If there is *exactly* one clause with $|S_j \cap U| \geq 2$ then c and w have the same score and c is not a unique winner of the final round, and if there are several clauses with $|S_j \cap U| \geq 2$ then c is not even a winner of the final round. Furthermore, every candidate u_i again has fewer points than w. In total, (C, V, c) is a no-instance of veto-CCPV-TP. □

By changing the distinguished candidate in this proof from c to w we obtain the following corollary to the above proof.

Corollary 1 *Veto-*DCPV-TP *is* NP*-complete in the nonunique-winner model.*

With a slight modification in the construction of the proof of Theorem 3 we can show the following result.

Theorem 4 *Veto-*CCPV-TP *is* NP*-complete in the nonunique-winner model.*

Proof Membership of veto-CCPV-TP in NP again is clear. For proving NP-hardness, we use the construction from the proof of Theorem 3 that maps a given ONE-IN-THREE-POSITIVE-3SAT instance (X, S), where $X = \{x_1, \ldots, x_m\}$ and $S = \{S_1, \ldots, S_n\}$ with $n > 1$, to the election (C, V), except that now we have n instead of $n - 1$ votes of the form $w \cdots c$. Again, let c be the distinguished candidate.

We show that (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, c) is a yes-instance of veto-CCPV-TP in the nonunique-winner model.

($\Rightarrow$) Starting from a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, we use the same partition of V as in the proof of Theorem 3. We again have that c, w, and all candidates

from $U \subseteq X$ with $|U \cap S_j| = 1$ for each clause S_j take part in the final round. However, c and w now have the same number n of vetoes, so they both are winners. Hence, (C, V, c) is a yes-instance of veto-CCPV-TP in the nonunique-winner model.

($\Leftarrow$) Note that the additional veto for c ensures that w now is the only winner of any possible final-round election, since w takes part in every final round and, while we have seen in the proof of Theorem 3 that w is a winner of every final round, the additional veto for c (i.e., the additional point for w) now ensures that w is the unique winner of every final round. It follows that a given no-instance of ONE-IN-THREE-POSITIVE-3SAT is mapped to a no-instance of veto-CCPV-TP in the nonunique-winner model. □

As before, by changing the distinguished candidate in this proof from c to w we obtain the following corollary to the above proof.

Corollary 2 *Veto-*DCPV-TP *is* NP*-complete in the unique-winner model.*

4 Control by partition of candidates in veto elections

We now turn to control by partition of candidates in veto elections, considering both constructive and destructive control, both tie-handling models, TE and TP, both the unique-winner and the nonunique-winner model, and the partition problems both with and without run-off. We start with constructive control by (run-off) partition of candidates.

The construction used to prove Theorem 5 is similar to the construction in the proofs of Theorems 3 and 4 in terms of the voters' preferences in the election defined from a given instance of ONE-IN-THREE-POSITIVE-3SAT; however, it differs in terms of the number of votes with these preferences. Note that one and the same construction suffices to prove resistance for many cases of control by partition of candidates while the argument of correctness has to be suitably tailored to each case.

Theorem 5 1. *Veto-*CCRPC-TP *and veto-*CCRPC-TE *are* NP*-complete in the unique-winner and the nonunique-winner model.*
2. *Veto-*CCPC-TP *and veto-*CCPC-TE *are* NP*-complete in the unique-winner model.*

Proof Membership of all problems in NP is obvious. To prove their NP-hardness, we reduce from ONE-IN-THREE-POSITIVE-3SAT and let (X, S) be an instance of this problem, where $X = \{x_1, \ldots, x_m\}$ and $S = \{S_1, \ldots, S_n\}$ with $n > 1$. Construct an election (C, V) with distinguished candidate $c \in C$ by defining $C = X \cup \{c, w\}$, where the elements of X from now on will also be viewed as candidates, and the list V of votes as follows:

number of votes	preference	
$3n^2 + 1 + i(2n^2 + 4n)$	$w\ c\ \cdots\ x_i$	for each $i \in \{1, \ldots, m\}$
$n - 1$	$w\ \cdots\ c$	
1	$c\ \cdots\ w\ S_j \setminus \{x_i\}$	for each $j \in \{1, \ldots, n\}$ and $x_i \in S_j$
$2n$	$w\ \cdots\ c\ S_j$	for each $j \in \{1, \ldots, n\}$

The reduction can be computed in polynomial time. In total, we have $m + 2$ candidates and

$$2n^2+4n-1+\sum_{i=1}^{m}\Big(3n^2+1+i(2n^2+4n)\Big) = 2n^2+4n-1+m(3n^2+1)+(n^2+2n)(m^2+m)$$

voters. Let $Q = \sum_{i=1}^{m} 3n^2 + 1 + i(2n^2 + 4n) = m(3n^2 + 1) + (n^2 + 2n)(m^2 + m)$, which is the number of voters in the first group. Candidate w alone wins in election (C, V), since the candidates score the following points:

$$\begin{aligned} score(c) &= Q + 3n + 2n^2, \\ score(w) &= Q + n - 1 + 3n + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 + 4n + 1) + n - 1 + 3n + 2n^2, \quad 1 \leq i \leq m. \end{aligned}$$

To prove the first item of the theorem, it remains to show that (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, c) is a yes-instance of veto-CCRPC, regardless of the tie-handling rule and the winner model used.

($\Rightarrow$) If (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset $U = \{u_1, \ldots, u_\ell\}$ of X (renaming its elements for convenience) such that $|U \cap S_j| = 1$ for each $j \in \{1, \ldots, n\}$.

We claim that partitioning C into $C_1 = U \cup \{c, w\}$ and $C_2 = C \setminus C_1$ ensures that c becomes the only veto winner of the run-off. To see this, note that the candidates in subelection (C_1, V) have the following scores:

$$\begin{aligned} score(c) &= Q + 3n + 2n^2, \\ score(w) &= Q + n - 1 + 2n + 2n^2, \text{ and} \\ score(u_i) &\leq Q - (5n^2 + 4n + 1) + n - 1 + 3n - 2 + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

(If we would allow that there are variables not occurring in any clause, we would have $score(u_i) \leq Q - (5n^2 + 4n + 1) + n - 1 + 3n + 2n^2$, $1 \leq i \leq \ell$. However, without loss of generality, we exclude this case.)

Since c alone wins in subelection (C_1, V), c will move forward to the final run-off. In the other subelection, (C_2, V), the candidate from $X \setminus U$ with the smallest subscript, say x_i, is the unique winner, because this candidate receives at most $3n^2+1+i(2n^2+4n)+3n+2n^2$ vetoes while every other candidate has at least $3n^2 + 1 + (i + 1)(2n^2 + 4n)$ vetoes, thus at least n vetoes more.

Therefore, the final run-off is $(\{c, x_i\}, V)$. Candidate c can get at most $2n^2 + n - 1$ vetoes from the second and the fourth voter groups, which is less than the number of vetoes x_i must get from the first and third voter groups. It follows that c alone wins the run-off. Thus (C, V, c) is a yes-instance of veto-CCRPC, regardless of the tie-handling rule and the winner model used.

($\Leftarrow$) Conversely, let (X, S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. Then, for any partition of X into X_1 and X_2, let k_i be the number of clauses containing i literals from X_1, $0 \leq i \leq 3$. We have $k_0 + k_1 + k_2 + k_3 = n$ and $k_1 \neq n$, since we started from a no-instance of ONE-IN-THREE-POSITIVE-3SAT. We will show that for each possible combination of the k_i (corresponding to each possible partition of X), candidate c cannot end up being the only veto winner. Note that a partition of X induces a partition of $C = X \cup \{c, w\}$ into C_1 and $C_2 = C \setminus C_1$ (assuming, without loss of generality, that

$c \in C_1$). It is enough to distinguish the three cases below, and in each case, we will show that c is not the only veto winner.

Case 1: $C_1 = \{c, w\}$. Then $score(c) = 3n$ and $score(w) = Q + n - 1 + 2n^2$, so w is the only veto winner of this subelection, and since c does not take part in the final run-off, c will not be an overall winner.

Case 2: C_1 contains c and some elements of X but not w. The only winner of subelection (C_1, V) is c, because c gets at most $n - 1 + 2n^2$ vetoes while all the remaining candidates $x_i \in C_1$ get at least $3n^2 + 1 + 2n^2 + 4n$ vetoes from the voters of the first group. In the other subelection, w wins alone, because w gets at most $3n$ vetoes while again every other candidate $x_i \in C_2$ gets at least $3n^2 + 1 + 2n^2 + 4n$ vetoes from the voters of the first group. Therefore, only c and w participate in the final run-off and, as we have seen in Case 1, w wins alone, which precludes c's overall victory in this case.

Case 3: C_1 contains c, w, and some elements of X. Distinguish the following three subcases.

Case 3.1: $k_0 \geq 2$. In this case, we have the following bounds on the scores of the candidates in their subelection:

$$\begin{aligned} score(c) &\leq Q + 3n + 2n^2 - 4n, \\ score(w) &\geq Q + n - 1 + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 - 4n - 1) + n - 1 + 3n + 2n^2, \quad x_i \in C_1. \end{aligned}$$

Candidate w has at least $2n - 1$ points more than c. Therefore, c does not reach the final run-off and cannot win.

Case 3.2: $k_0 = 1$. In this case, we have the following bounds on the scores of the candidates in their subelection:

$$\begin{aligned} score(c) &= Q + 3n + 2n^2 - 2n, \\ score(w) &\geq Q + n - 1 + 2n - 2 + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 - 4n - 1) + n - 1 + 3n + 2n^2, \quad x_i \in C_1. \end{aligned}$$

Now, the inequality $3 < 2n$ (which is true for $n > 1$) implies $score(w) > score(c)$ also in this case. Also, w defeats every candidate $x_i \in C_1$ due to the $5n^2 - n - 1$ vetoes that every candidate x_i receives already from the voters of the first group.

Case 3.3: $k_0 = 0$. In this case, we have the following bounds on the scores of the candidates in their subelections:

$$\begin{aligned} score(c) &= Q + 3n + 2n^2, \\ score(w) &\geq Q + n - 1 + 2n + 1 + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 - 4n - 1) + n - 1 + 3n + 2n^2, \quad x_i \in C_1. \end{aligned}$$

Candidates c and w have the same score if and only if there is exactly one clause S_j with $|S_j \cap C_1| \geq 2$. If we use TE, since both c and w win their subelection (C_1, V), no one can move forward to the final run-off, and it follows that c will not be an overall winner. If we use TP, both c and w will move forward to the final run-off. In the other subelection, (C_2, V), the candidate from C_2 with the smallest subscript in this subelection wins alone,

say x_i. Therefore, c, w, and x_i face each other in the final run-off. Since we started from a no-instance of ONE-IN-THREE-POSITIVE-3SAT, there exists at least one clause S_j such that $S_j \cap \{x_i\} = \emptyset$. Thus the same as in Case 3.1 or in Case 3.2 applies, and w is the only winner and c does not win.

If $|S_j \cap C_1| \geq 2$ holds true for more than one clause S_j, w wins subelection (C_1, V) alone, so c cannot move forward to the final run-off and thus cannot win.

Thus, in model TP, each possible final round occurs in one of the cases we just considered and always contains w. This also holds in model TE except when $|S_j \cap C_1| \geq 2$ for exactly one clause S_j, and since in this case both c and w win their subelection, they block each other from moving to the final round.

As we have shown that c is not a veto winner for any partition of the candidates, (C, V, c) is a no-instance of veto-CCRPC, regardless of the tie-handling rule and the winner model used. This completes the proof of the first item of the theorem.

To prove the second item of the theorem, we now show that (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, c) is a yes-instance of veto-CCPC, regardless of the tie-handling rule, in the unique-winner model.

($\Rightarrow$) If (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset U of X such that $|U \cap S_j| = 1$ for each $j \in \{1, \ldots, n\}$. Partition C into $C_1 = \{c, w\} \cup U$ and $C_2 = C \setminus C_1$. As we have shown in the proof of ($\Rightarrow$) for the first item of this theorem, c is the only winner of the subelection (C_1, V). In the final round, c now faces each candidate $x_i \in C_2$. Recall that $Q = m(3n^2 + 1) + (n^2 + 2n)(m^2 + m)$ is the number of voters of the first group. For each $x_i \in C_2$, we have

$$score(c) \geq Q + 3n > Q - (3n^2 + 1 + 2n^2 + 4n) + n - 1 + 2n^2 \geq score(x_i).$$

Since c gets the most points, c is the only winner of the final round. Thus (C, V, c) is a yes-instance of veto-CCPC, regardless of the tie-handling rule, in the unique-winner model.

($\Leftarrow$) Conversely, let (X, S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. We have to show that (C, V, c) is a no-instance of veto-CCPC as well, regardless of the tie-handling rule, in the unique-winner model. In the proof of ($\Leftarrow$) for the first item of this theorem, we have seen that w wins her subelection for every partition of candidates. Only in Case 3.3, where c, w, and some candidates x_i participate in one first-round subelection and $k_0 = 0$, w is not a *unique* winner and does not proceed to the final round if we use model TE, but then c does not reach the final round either. In all other cases, w reaches and wins the final round, so c cannot be a unique winner. Hence, (C, V, c) is a no-instance of veto-CCPC, regardless of the tie-handling rule, in the unique-winner model. This completes the proof of the theorem. □

By changing the distinguished candidate from c to w in the above proof we have the following corollary. In particular, when we use model TP, note that w always reaches and wins the final round in the proof of ($\Leftarrow$).

Corollary 3 *Veto-*DCRPC-TP *is* NP*-complete in the unique-winner and the nonunique-winner model.*[7]

[7]Note that DCRPC-TP and DCPC-TP are known to be identical problems in the nonunique-winner model for all voting systems [23, Thm. 8 on p. 386]; the proofs can be found in the related technical report by Hemaspaandra et al. [22]. Thus veto-DCPC-TP is NP-complete in the nonunique-winner model as well.

A minor tweak in the construction of the proof of Theorem 5 (namely, by having n instead of $n-1$ votes of the form $w \cdots c$, all else being equal) works for showing NP-hardness of veto-CCPC-TE and veto-CCPC-TP in the nonunique-winner model and of veto-DCRPC-TE in both winner models. Again, note that one and the same construction suffices to prove resistance for all these cases of control by partition of candidates while the argument of correctness will be suitably tailored to each case.

Theorem 6 1. *Veto-*CCPC-TE *and veto-*CCPC-TP *are* NP*-complete in the nonunique-winner model.*

2. *Veto-*DCRPC-TE *is* NP*-complete in both the unique-winner and the nonunique-winner unique-winner and the nonunique-winner model.*[8]

Proof Membership of all problems in NP is again obvious. To prove their NP-hardness, we again reduce from ONE-IN-THREE-POSITIVE-3SAT. Let (X, S) be an instance of ONE-IN-THREE-POSITIVE-3SAT with $X = \{x_1, \ldots, x_m\}$ and $S = \{S_1, \ldots, S_n\}$, $n > 1$. Construct an election (C, V) with distinguished candidate $c \in C$ by defining $C = X \cup \{c, w\}$, where the elements of X from now on will also be viewed as candidates, and the list V of votes as follows:

number of votes	preference	
$3n^2 + 1 + i(2n^2 + 4n)$	$w\ c\ \cdots\ x_i$	for each $i \in \{1, \ldots, m\}$
$n - 1$	$w\ \cdots\ c$	
1	$c\ \cdots\ w\ S_j \setminus \{x_i\}$	for each $j \in \{1, \ldots, n\}$ and $x_i \in S_j$
$2n$	$w\ \cdots\ c\ S_j$	for each $j \in \{1, \ldots, n\}$

The reduction is polynomial-time computable. There are $m + 2$ candidates and

$$2n^2 + 4n + \sum_{i=1}^{m} \Big(3n^2 + 1 + i(2n^2 + 4n)\Big) = 2n^2 + 4n + m(3n^2 + 1) + (n^2 + 2n)(m^2 + m)$$

voters. As in the proof of Theorem 5, let $Q = m(3n^2 + 1) + (n^2 + 2n)(m^2 + m)$ be the number of voters from the first group. Observe that w is the only veto winner of election (C, V):

$$\begin{aligned} score(c) &= Q + 3n + 2n^2 \\ score(w) &= Q + n + 3n + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad 1 \leq i \leq m. \end{aligned}$$

To prove the first item of the theorem, we show that (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, c) is a yes-instance of veto-CCPC, regardless of the tie-handling rule, in the nonunique-winner model.

($\Rightarrow$) If (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset U of X such that $|U \cap S_j| = 1$ for each $j \in \{1, \ldots, n\}$. Partition C into $C_2 = U \cup \{c\}$ and $C_1 = C \setminus C_2$. In subelection (C_1, V), for each $x_i \in C_1$, we have

$$score(w) = Q + n + 3n + 2n^2 > Q - 3n^2 - 1 - 2n^2 - 4n + n + 3n + 2n^2 \geq score(x_i).$$

[8] In both winner models, the problems DCRPC-TE and DCPC-TE are known to be identical for all voting systems [23, Thm. 8 on p. 386]. Thus veto-DCPC-TE is NP-complete in both winner models as well.

Thus w gets all possible points from the votes of the form $c \cdots w\ S_j \setminus \{x_i\}$ for each $j \in \{1, \ldots, n\}$ and $x_i \in S_j$ because every S_j contains exactly two candidates of $X \setminus U$. Since w receives the most points in this subelection, w is its only winner and moves forward to the final round. In the final round, there are w and all candidates from C_2. They have the following scores:

$$\begin{aligned} score(c) &= Q + 3n + 2n^2, \\ score(w) &= Q + n + 2n + 2n^2, \text{ and} \\ score(x_i) &\leq Q - 5n^2 - 4n - 1 + n + 3n + 2n^2, \quad x_i \in C_2. \end{aligned}$$

Both c and w have the highest score and are the overall veto winners. Thus (C, V, c) is a yes-instance of veto-CCPC, regardless of the tie-handling rule, in the nonunique-winner model.

($\Leftarrow$) Conversely, let (X, S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. As in the proof of Theorem 5, we consider all possible partitions of C into C_1 and C_2 and show that c is never a veto winner overall. To this end, it is sufficient to show that w joins the final round for each partition of C into C_1 and C_2.

Case 1: $C_1 = \{c, w\}$. Then $score(c) = 3n$ and $score(w) = Q + n + 2n^2$, so w is the only veto winner of this subelection, and since c does not take part in the final round, c will not be an overall winner.

Case 2: C_1 contains w but not c. Then we have

$$\begin{aligned} score(w) &\geq Q + n + 2n^2 \text{ and} \\ score(x_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad x_i \in C_1. \end{aligned}$$

Now, the inequality $0 < 5n^2 + n + 1$ implies $score(w) > score(x_i)$ also in this case.

Case 3: C_1 contains c, w, and some elements of X. Let k be the number of clauses S_j such that $|S_j \cap U| \neq 0$. Then the scores in (C_1, V) are:

$$\begin{aligned} score(c) &= Q + 3n + 2kn, \\ score(w) &\geq Q + n + 2k + 2n^2, \text{ and} \\ score(x_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad x_i \in C_1. \end{aligned}$$

For w to win subelection (C_1, V) alone, we need to show that $score(w) > score(c)$ and $score(w) > score(x_i)$ for each $x_i \in X$. Simplifying the scores of c and w, we get $2k+2n^2 > 2n+2kn$, which is equivalent to $n+\frac{k}{n} > 1+k$, which for $n > 1$ in turn is true for $k < n$. For $k = n$, there must be a clause S_j that contains at least two literals; otherwise, we would have started from a yes-instance of ONE-IN-THREE-POSITIVE-3SAT. But then we have that

$$score(w) \geq Q + n + 2(n-1) + 3 + 2n^2 > Q + 3n + 2n^2 = score(c).$$

On the other hand, w also wins out over each $x_i \in X$, since simplifying their scores yields $n + 2k + 2n^2 > -3n^2$, which is true for each n.

Case 4: C_2 contains w. Then w is immediately in the final round.

Thus, in each case, w takes part in the final round and is the unique veto winner of the final round for any partition of the candidates. It follows that (C, V, c) is a no-instance of veto-CCPC, regardless of the tie-handling rule, in the nonunique-winner model, which completes the proof of the first item of the theorem.

To prove the second item of the theorem, we now show that (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, w) is a yes-instance of veto-DCRPC-TE in both winner models.

($\Rightarrow$) If (X, S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT, then there is a subset U of X such that $|U \cap S_j| = 1$ for each $j \in \{1, \ldots, n\}$. Partitioning C into $C_1 = U \cup \{c, w\}$ and $C_2 = C \setminus C_1$ ensures that w is not the only veto winner, since c and w have the same score in subelection (C_1, V):

$$\begin{aligned} score(c) &= Q + 3n + 2n^2 \text{ and} \\ score(w) &= Q + n + 2n + 2n^2; \end{aligned}$$

so, by model TE, w cannot move forward to the final round. Thus (C, V, w) is a yes-instance of veto-DCRPC-TE in both winner models.

($\Leftarrow$) Conversely, let (X, S) be a no-instance of ONE-IN-THREE-POSITIVE-3SAT. As in the previous proofs, we consider all possible partitions of C into C_1 and C_2 and show that w always is the only veto winner of the resulting run-off.

Case 1: $C_1 = \{c, w\}$. Then $score(w) = Q + n + 2n^2$ and $score(c) = 3n$, so w moves forward to the final round. If the other subelection, (C_2, V), has more than one winner, TE blocks them all, so w wins. If (C_2, V) has a unique winner, say x_i, we have $score(w) \geq Q + n + 2n^2$ and $score(x_i) \leq 3n$ in the final round, $(\{w, x_i\}, V)$, so w wins.

Case 2: C_1 contains w but not c. Then

$$\begin{aligned} score(w) &\geq Q + n + 2n^2 \text{ and} \\ score(x_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad x_i \in C_1, \end{aligned}$$

implies that w scores at least $5n^2 + n + 1$ points more than any $x_i \in C_1$ and moves forward to the final run-off. If subelection (C_2, V) has more than one winner, w outright wins the run-off; if either c or some x_i wins in (C_2, V), w alone wins the run-off as shown in Case 1.

Case 3: C_1 contains c, w, and some elements of X. Rename the elements of $U = C_1 \cap X$ by $U = \{u_1, \ldots u_\ell\}$ for convenience. Let k be the number of clauses S_j such that $|S_j \cap U| = 0$.

Case 3.1: $k > 0$. Then the scores of the candidates in (C_1, V) are bounded as follows:

$$\begin{aligned} score(c) &= Q + 3n + 2n(n - k), \\ score(w) &\geq Q + n + 2(n - k) + 2n^2, \text{ and} \\ score(u_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

For w to win subelection (C_1, V) alone, we need to show that $score(w) > score(c)$ and $score(w) > score(u_i)$ for each $u_i \in U$. For the former inequality, we have

$$score(w) \geq Q + 3n - 2k + 2n^2 > Q + 3n + 2n^2 - 2nk = score(c),$$

which is equivalent to $2nk > 2k$, which in turn is true because $k > 0$ and $n > 1$. For the latter inequality, w also wins out over each $u_i \in U$, since simplifying their scores:

$$score(w) \geq Q + n + 2(n - k) + 2n^2 > Q - (5n^2 + 4n + 1) + n + 3n + 2n^2 \geq score(u_i)$$

yields $5n^2 + 3n + 1 > 2k$, which is true because $k \leq n$. Since w alone wins subelection (C_1, V), w moves forward to the final run-off. In the run-off, w is either alone or faces

some x_i (if x_i is the only veto winner of subelection (C_2, V)). By the argument just given, w triumphes over x_i and is the only overall veto winner.

Case 3.2: $k = 0$. Since (X, S) is a no-instance of ONE-IN-THREE-POSITIVE-3SAT, there is at least one clause S_j with $|S_j \cap U| \geq 2$ in this case. This implies the following bounds on the scores in (C_1, V):

$$\begin{aligned} score(c) &= Q + 3n + 2n^2, \\ score(w) &\geq Q + n + 2n + 1 + 2n^2, \text{ and} \\ score(u_i) &\leq Q - (5n^2 + 4n + 1) + n + 3n + 2n^2, \quad 1 \leq i \leq \ell. \end{aligned}$$

Thus w is the only veto winner of subelection (C_1, V) and (by the above arguments) wins also the final run-off alone. Hence, (C, V, w) is a no-instance of veto-DCRPC-TE, regardless of the winner model. This completes the proof of the theorem. □

The only case still missing is veto-DCPC-TP in the unique-winner model. This case immediately follows from the proof of the first part of Theorem 6 by changing the distinguished candidate from c to w.

Corollary 4 *Veto-*DCPC-TP *is* NP*-complete in the unique-winner model.*

5 Constructive control by adding candidates in plurality elections

In this section, we consider only a single control scenario (constructive control by adding candidates) for the simplest natural voting system, plurality. That plurality is resistant to this control type in the sense of plurality-CCAC being NP-hard has already been known since the first paper on electoral control, due to Bartholdi et al. [2]. Recently, Chen et al. [8] considered the parameterized complexity of control problems for natural voting systems when there are only few voters. In particular, they proved that the parameterized variant of plurality-CCAC (the problem formalizing constructive control by adding candidates, as defined below), parameterized by the number of voters, is W[1]-hard by reducing from the W[1]-hard problem MULTI-COLORED-CLIQUE, parameterized by the clique order [8, Theorem 1]. However, while the proof sketch of this result does provide a very clever reduction, it is technically flawed. In this section, we first briefly present their reduction from the proof sketch of [8, Theorem 1], then give a counterexample showing that it is not correct, and finally fix this flaw by suitably adapting their reduction in order to make it correct.

The W[1]-hard parameterized problem Chen et al. [8] reduce from is formally defined as follows:

Multi-Colored-Clique

Given:	An undirected graph $G = (V(G), E(G))$, where $V(G)$ is partitioned into h sets $V_1(G), \ldots, V_h(G)$ such that each $V_i(G) = \left\{v_1^{(i)}, \ldots, v_{n'}^{(i)}\right\}$ consists of exactly n' vertices with color i and G has only edges connecting vertices of distinct colors.
Parameter:	The number h of colors.
	Does there exist a size-h clique containing some vertex for each color?

Given a voting rule $\mathscr{E}$, the parameterized problem $\mathscr{E}$-CCAC (parameterized by the number of voters) is defined as follows.

$\mathscr{E}$-CCAC	
Given:	A set C of registered candidates, a set A of as yet unregistered candidates, $C \cap A = \emptyset$, a list of preferences V over $C \cup A$, a nonnegative integer k, and a distinguished candidate $p \in C$.
Parameter:	The number of votes in V.
Question:	Does there exist a subset $A' \subseteq A$ such that $\|A'\| \leq k$ and p is an $\mathscr{E}$ winner of the election $(C \cup A', V')$ with V' being V restricted to $C \cup A'$?

We now describe the reduction from the proof sketch of Theorem 1 due to Chen et al. [8]. Let $G = (V(G), E(G))$ be a given undirected graph, where $V(G)$ is partitioned into h sets $V_1(G), \ldots, V_h(G)$ such that each $V_i(G) = \left\{v_1^{(i)}, \ldots, v_{n'}^{(i)}\right\}$ consists of exactly n' vertices with color i and G has only edges connecting vertices of distinct colors. Construct the following instance (C, A, V, k, p) of plurality-CCAC:

- The set of registered candidates is $C = \{p, d\}$, where p is the distinguished candidate the chair wants to see win.
- The set A of unregistered candidates contains
  - a *vertex candidate* v for each $v \in V(G)$ and
  - two *edge candidates* (u, v) and (v, u) for each edge $\{u, v\} \in E(G)$.
- To specify the list V of votes, we adopt the following notation from [8]. Let $E(i, j)$ be the set of all edge candidates (u, v) with $u \in V_i(G)$ being colored i and $v \in V_j(G)$ being colored j.

  For each vertex $v_z^{(i)} \in V_i(G)$, let $L(v_z^{(i)}, j)$ be the set of all edge candidates $(v_z^{(i)}, v)$ with $v \in V_j(G)$ and $(v_z^{(i)}, v) \in E(G)$.

  For each i, j, $1 \leq i \neq j \leq n$, define the following two linear orders:

$$R(i, j) : v_1^{(i)}\ L(v_1^{(i)}, j)\ \cdots\ v_{n'}^{(i)}\ L(v_{n'}^{(i)}, j),$$
$$R'(i, j) : L(v_1^{(i)}, j)\ v_1^{(i)}\ \cdots\ L(v_{n'}^{(i)}, j)\ v_{n'}^{(i)}.$$

  Now we are ready to define the following three types of votes:

  1. For each i, there is one vote of the form $v_1^{(i)}\ \cdots\ v_{n'}^{(i)}\ d\ \cdots$.
  2. For each pair of colors i, j, $1 \leq i \neq j \leq n$, there are
     (a) $h - 1$ votes of the form $E(i, j)\ d\ \cdots$,
     (b) one vote of the form $R(i, j)\ d\ \cdots$, and
     (c) one vote of the form $R'(i, j)\ d\ \cdots$.
  3. There are h votes of the form $d\ \cdots$ and h votes of the form $p\ \cdots$.
- At most $k = h + 2\binom{h}{2}$ candidates can be added.

Chen et al. [8] then argue that p can become a plurality winner by adding at most k candidates from A if and only if graph G has a size-h multi-colored clique (i.e., a clique containing a vertex for each color). However, we now present a counterexample for this claim.

Example 2 Figure 1 shows a graph G corresponding to a no-instance of MULTI-COLORED-CLIQUE. In particular, the vertex set $V(G)$ is partitioned into three sets containing two vertices each:

$$V_1(G) = \{1, 6\}, \quad V_2(G) = \{2, 5\}, \quad V_3(G) = \{3, 4\}$$

but G has no clique of size three.

However, we now show that the above construction maps this no-instance of MULTI-COLORED-CLIQUE to a yes-instance of plurality-CCAC.

Indeed, from G we obtain the set $C = \{p, d\}$ of registered candidates, where p is the distinguished candidate the chair wants to see win. The set of unregistered candidates is

$$A = \left\{1, 2, 3, 4, 5, 6, \begin{matrix}2\\1\end{matrix}, \begin{matrix}1\\2\end{matrix}, \begin{matrix}3\\2\end{matrix}, \begin{matrix}2\\3\end{matrix}, \begin{matrix}4\\2\end{matrix}, \begin{matrix}2\\4\end{matrix}, \begin{matrix}6\\4\end{matrix}, \begin{matrix}4\\6\end{matrix}, \begin{matrix}6\\5\end{matrix}, \begin{matrix}5\\6\end{matrix}\right\}$$

with six vertex candidates and ten edge candidates. Figure 2 gives the list V of votes over $C \cup A$, where the number before a vote indicates how many votes of this type there are according to the above construction. Finally, since $h = 3$, we are allowed to add $k = 3 + 2\binom{3}{2} = 9$ candidates.

Since G has no clique of size three, it should be impossible to make p a plurality winner by adding at most $k = 9$ candidates. However, adding the candidates

$$6, 2, 4, \begin{matrix}5\\6\end{matrix}, \begin{matrix}1\\2\end{matrix}, \begin{matrix}4\\6\end{matrix}, \begin{matrix}3\\2\end{matrix}, \begin{matrix}6\\4\end{matrix}, \begin{matrix}2\\4\end{matrix}$$

to the election implies that each candidate scores exactly three points. In particular, p has become a plurality winner, which shows that a no-instance of MULTI-COLORED-CLIQUE has been mapped to a yes-instance of plurality-CCAC by the reduction presented in the proof sketch of [8, Theorem 1].

Let us discuss the observation made in Example 2 in general and let us see how the reduction can be adapted so as to work correctly.

First note that p can never score more than h points, no matter how many candidates are added to the election. Thus, for p to be a winner, no other candidate must score more than h points. Candidate d will score at least h points, no matter if any (and how many) candidates are added. To prevent d from scoring more than h points, any size-$\left(h + 2\binom{h}{2}\right)$ set $A' \subseteq A$ of added unregistered candidates must contain exactly one vertex candidate for each color and exactly one edge candidate of each (ordered) pair of colors. In their proof sketch, Chen et al. [8, p. 2049] further say that "if A' contains two vertex candidates u, v but not the edge candidate (u, v) then, due to the orders $R(i, j) \succ d \succ \cdots$ and $R'(i, j) \succ d \succ \cdots$, either u or an edge candidate (u', v') (where $u' \in V_i(G)$, $v' \in V_j(G)$, but $(u', v') \neq (u, v)$)

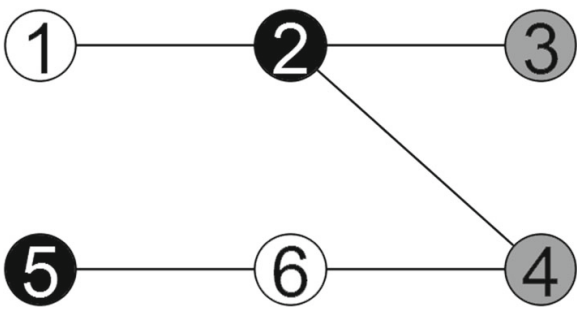

Fig. 1 Counterexample for the reduction for [8, Theorem 1]

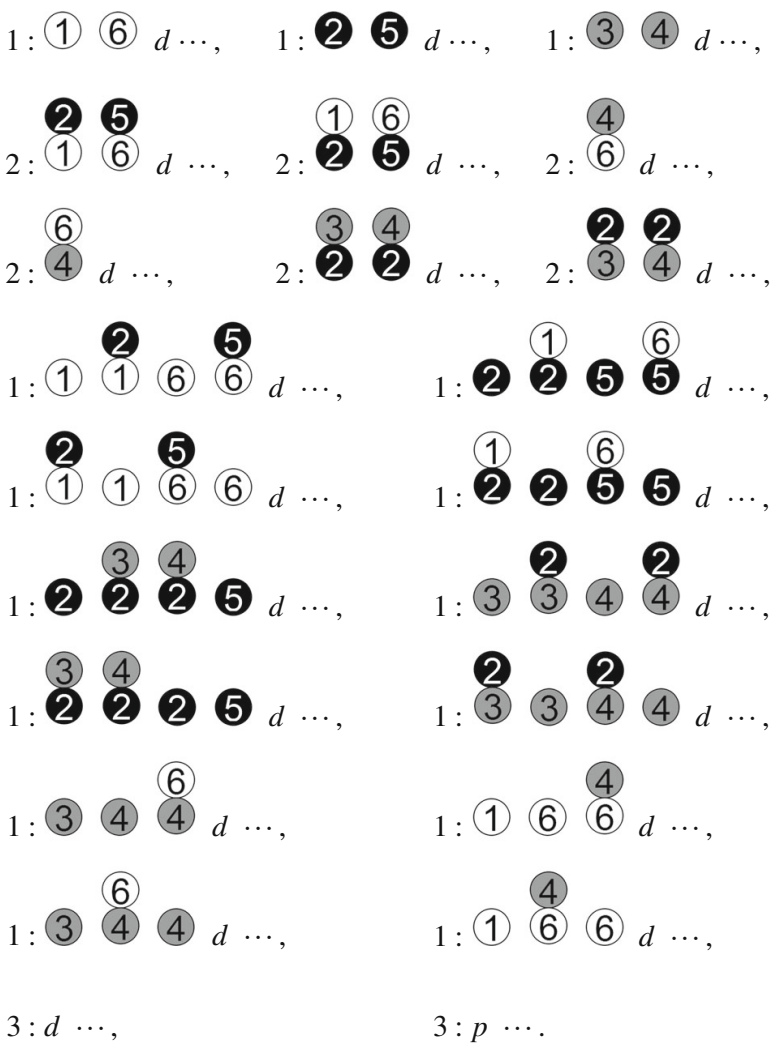

Fig. 2 Constructing a yes-instance of plurality-CCAC from a no-instance of MULTI-COLORED-CLIQUE according to the proof sketch of [8, Theorem 1]

receives too many points, causing p not to win."[9] That is, they claim that adding two vertex candidates, $u \in V_i(G)$ and $v \in V_j(G)$, enforces addition of the edge candidate (u, v) because this would be required to ensure that the points from the two votes $R(i, j)\ d\ \cdots$ and $R'(i, j)\ d\ \cdots$ are scored by *different* candidates. This, however, is not always true. Indeed, to ensure that different candidates score points from these two votes, it is enough to add with u and v an edge candidate (u, v_j) such that v_j and v have the same color and $(u, v_j) \in E(G)$. Therefore, it in fact is possible to add two edge candidates (u, v) and (v', u') with $u, u' \in V_i(G)$, $v, v' \in V_j(G)$, $i \neq j$, and $(u, v) \neq (u', v')$.

In Example 2, the vertex candidates ❷ and ④ and also the edge candidate $\overset{3}{❷}$ have been added. The problem is that one is not forced to also add the edge candidate $\overset{2}{③}$. The above argument is only correct if one assumes that the edge candidates (u, v) and (v, u) must always be added together. To enforce this, one can adapt the votes $E(i, j)\ d\ \cdots$ by requiring

[9] Note that we omit the order symbol $\succ$, so the orders $R(i, j) \succ d \succ \cdots$ and $R'(i, j) \succ d \succ \cdots$ in this quote are written $R(i, j)\ d\ \cdots$ and $R'(i, j)\ d\ \cdots$ here.

each edge candidate (u, v) is followed by the corresponding edge candidate (v, u). For instance, in Example 2 that means that the two votes

are both changed to

Now, if one adds two *unmatching* edge candidates (i.e., (u, v) and (v', u') with $(u, v) \neq (u', v')$, $u, u' \in V_i$, and $v, v' \in V_j$) then, without loss of generality, edge candidate (u, v) receives the points in the now modified votes

$$E(i, j)\ d\ \cdots = \cdots\ (u, v)\ (v, u)\ \cdots (u', v')\ (v', u')\ \cdots\ d\ \cdots$$

and

$$E(j, i)\ d\ \cdots = \cdots\ (v, u)\ (u, v)\ \cdots (v', u')\ (u', v')\ \cdots\ d\ \cdots .$$

However, if one adds the *matching* edge candidates, say (u, v) and (v, u), then each of them receives a point only from one of these modified votes. This enforces that only *matching* edge candidates can be added (otherwise, p would not win). The votes

$$\begin{aligned} R'(i, j):&\quad v_1^{(i)}\ L(v_1^{(i)}, j)\ \cdots\ v_{n'}^{(i)}\ L(v_{n'}^{(i)}, j), \\ R'(i, j):&\quad L(v_1^{(i)}, j)\ v_1^{(i)}\ \cdots\ L(v_{n'}^{(i)}, j)\ v_{n'}^{(i)} \end{aligned}$$

then imply that with an edge candidate (u, v) also the candidate u must be added. If some other vertex candidate $u' \neq u$ were added, the above two votes restricted to candidates u' and (u, v) would either both be $u'\ (u, v)$ or both be $(u, v)\ u'$, which would give one of these two candidates too many points.

Observe that matching vertex and edge candidates can be added only if there is a size-h multi-colored clique in the given graph G. If there is no such clique, at least one unmatching candidate has to be added.

6 Conclusions and future research

We have studied the complexity of control by partition of either voters or candidates for veto elections. We have pinpointed the complexity of all related cases in a broad variety of models, including all combinations of constructive versus destructive control, the ties-eliminate versus the ties-promote model, the unique-winner versus the nonunique-winner model, and for control by partition of candidates the cases with and without run-off.

Table 1 in the introduction gives an overview of all our results. It is interesting to note that, unlike for constructive coalitional weighted manipulation where veto and plurality behave quite differently,[10] the results obtained for control by partition in veto are the same as those known for control by partition in plurality [2, 24]: Control by partition of candidates is hard in all cases, whereas control by partition of voters is easy in the ties-eliminate model but hard in the ties-promote model.

[10]While this manipulation problem is easy to solve in plurality for any number of candidates, it is NP-complete in veto for three or more candidates [9]. Indeed, Hemaspaandra and Hemaspaandra [21] proved a dichotomy result saying that plurality is the only nontrivial scoring protocol for which constructive coalitional weighted manipulation is easy (and Conitzer et al. [9] observed this too for the case of three candidates).

On a higher level, a quite challenging interesting open question is to completely characterize the class of scoring protocols in terms of control complexity (i.e., to establish dichotomy results for the various control types), as has been done by Hemaspaandra et al. [28] for constructive control by adding voters, Hemaspaandra and Schnoor [29] for constructive control by deleting voters, by Hemaspaandra and Hemaspaandra [21] for constructive coalitional weighted manipulation, and by Betzler and Dorn [6] and Baumeister and Rothe [4] for the possible winner problem (a generalization of coalitional unweighted manipulation due to Konczak and Lang [30]). Finally, it would also be interesting to study veto with respect to the refined models of control by partition introduced by Erdélyi et al. [13] and by Puppe and Tasnádi [41].

Acknowledgements We thank the AMAI, AAMAS-2017, ECAI-2016, and ISAIM-2016 reviewers for many helpful suggestions. This work has been supported in part by DFG grant RO-1202/15-1.

References

1. Bartholdi, J. III, Tovey, C., Trick, M.: The computational difficulty of manipulating an election. Soc. Choice Welf. **6**(3), 227–241 (1989)
2. Bartholdi, J. III, Tovey, C., Trick, M.: How hard is it to control an election? Math. Comput. Model. **16**(8/9), 27–40 (1992)
3. Baumeister, D., Erdélyi, G., Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Computational aspects of approval voting. In: Laslier, J., Sanver, R. (eds.) Handbook on Approval Voting, chap. 10, pp. 199–251. Springer (2010)
4. Baumeister, D., Rothe, J.: Taking the final step to a full dichotomy of the possible winner problem in pure scoring rules. Inf. Process. Lett. **112**(5), 186–190 (2012)
5. Baumeister, D., Rothe, J.: Preference aggregation by voting. In: Rothe, J. (ed.) Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division, chap. 4, pp. 197–325. Springer (2015)
6. Betzler, N., Dorn, B.: Towards a dichotomy for the possible winner problem in elections based on scoring rules. J. Comput. Syst. Sci. **76**(8), 812–836 (2010)
7. Betzler, N., Uhlmann, J.: Parameterized complexity of candidate control in elections and related digraph problems. Theor. Comput. Sci. **410**(52), 5425–5442 (2009)
8. Chen, J., Faliszewski, P., Niedermeier, R., Talmon, N.: Elections with few voters: Candidate control can be easy. In: Proceedings of the 29th AAAI Conference on Artificial Intelligence, pp. 2045–2051. AAAI Press (2015)
9. Conitzer, V., Sandholm, T., Lang, J.: When are elections with few candidates hard to manipulate? Journal of the ACM **54**(3), Article 14 (2007)
10. Downey, R., Fellows, M.: Parameterized Complexity, 2nd edn. Springer-Verlag, Berlin (2013)
11. Elkind, E., Faliszewski, P., Slinko, A.: Cloning in elections: Finding the possible winners. J. Artif. Intell. Res. **42**, 529–573 (2011)
12. Erdélyi, G., Fellows, M., Rothe, J., Schend, L.: Control complexity in Bucklin and fallback voting: A theoretical analysis. J. Comput. Syst. Sci. **81**(4), 632–660 (2015)
13. Erdélyi, G., Hemaspaandra, E., Hemaspaandra, L.: More natural models of electoral control by partition. In: Proceedings of the 4th International Conference on Algorithmic Decision Theory, pp. 396–413. Springer-Verlag Lecture Notes in Artificial Intelligence #9346 (2015)
14. Erdélyi, G., Nowak, M., Rothe, J.: Sincere-strategy preference-based approval voting fully resists constructive control and broadly resists destructive control. Math. Log. Q. **55**(4), 425–443 (2009)
15. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.: How hard is bribery in elections J. Artif. Intell. Res. **35**, 485–532 (2009)
16. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.: Multimode control attacks on elections. J. Artif. Intell. Res. **40**, 305–351 (2011)
17. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.: Weighted electoral control. J. Artif. Intell. Res. **52**, 507–542 (2015)
18. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Llull and Copeland voting computationally resist bribery and constructive control. J. Artif. Intell. Res. **35**, 275–341 (2009)

19. Faliszewski, P., Rothe, J.: Control and bribery in voting. In: Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A. (eds.) Handbook of Computational Social Choice, chap. 7, pp. 146–168. Cambridge University Press (2016)
20. Garey, M., Johnson, D.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company, Stuttgart (1979)
21. Hemaspaandra, E., Hemaspaandra, L.: Dichotomy for voting systems. J. Comput. Syst. Sci. **73**(1), 73–83 (2007)
22. Hemaspaandra, E., Hemaspaandra, L., Menton, C.: Search versus decision for election manipulation problems. Tech. Rep. arXiv:1202.6641 [cs.GT], Computing Research Repository, arXiv.org/corr/ (2012)
23. Hemaspaandra, E., Hemaspaandra, L., Menton, C.: Search versus decision for election manipulation problems. In: Proceedings of the 30th Annual Symposium on Theoretical Aspects of Computer Science, LIPIcs, vol. 20, pp. 377–388. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2013)
24. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Anyone but him: The complexity of precluding an alternative. Artif. Intell. **171**(5–6), 255–285 (2007)
25. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Hybrid elections broaden complexity-theoretic resistance to control. Math. Log. Q. **55**(4), 397–424 (2009)
26. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: The complexity of controlling candidate-sequential elections. Theor. Comput. Sci. **678**, 14–21 (2017)
27. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: The complexity of online voter control in sequential elections. J. Auton. Agents Multi-Agent Syst. **31**(5), 1055–1076 (2017)
28. Hemaspaandra, E., Hemaspaandra, L., Schnoor, H.: A control dichotomy for pure scoring rules. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, pp. 712–720. AAAI Press (2014)
29. Hemaspaandra, E., Schnoor, H.: Dichotomy for pure scoring rules under manipulative electoral actions. In: Proceedings of the 22nd European Conference on Artificial Intelligence, pp. 1071–1079. IOS Press (2016)
30. Konczak, K., Lang, J.: Voting procedures with incomplete preferences. In: Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling, pp. 124–129 (2005)
31. Lin, A.: Solving hard problems in election systems. Ph.D. thesis, Rochester Institute of Technology, Rochester, NY, USA (2012)
32. Loreggia, A., Narodytska, N., Rossi, F., Venable, B., Walsh, T.: Controlling elections by replacing candidates or votes (extended abstract). In: Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems, pp. 1737–1738. IFAAMAS (2015)
33. Maushagen, C., Rothe, J.: Complexity of control by partitioning veto and maximin elections and of control by adding candidates to plurality elections. In: Proceedings of the 22nd European Conference on Artificial Intelligence, pp. 277–285. IOS Press (2016)
34. Maushagen, C., Rothe, J.: Complexity of control by partition of voters and of voter groups in veto and other scoring protocols. In: Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems, pp. 615–623. IFAAMAS (2017)
35. Menton, C.: Normalized range voting broadly resists control. Theory Comput. Syst. **53**(4), 507–531 (2013)
36. Neveling, N., Rothe, J.: Closing the gap of control complexity in Borda elections: Solving ten open cases. In: Proceedings of the 18th Italian Conference on Theoretical Computer Science. CEUR-WS.org (2017). To appear
37. Neveling, N., Rothe, J.: Solving seven open problems of offline and online control in Borda elections. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence, pp. 3029–3035. AAAI Press (2017)
38. Niedermeier, R.: Invitation to fixed-parameter algorithms. Oxford University Press, Oxford (2006)
39. Papadimitriou, C.: Computational Complexity, 2nd edn. Addison-Wesley, Boston (1995)
40. Parkes, D., Xia, L.: A complexity-of-strategic-behavior comparison between Schulze's rule and ranked pairs. In: Proceedings of the 26th AAAI Conference on Artificial Intelligence, pp. 1429–1435. AAAI Press (2012)
41. Puppe, C., Tasnádi, A.: Optimal redistricting under geographical constraints: Why "pack and crack" does not work. Econ. Lett. **105**(1), 93–96 (2009)
42. Rothe, J.: Complexity theory and cryptology. An Introduction to Cryptocomplexity. EATCS Texts in Theoretical Computer Science. Springer-Verlag, Berlin (2005)
43. Russel, N.: Complexity of control of Borda count elections. Master's thesis, Rochester Institute of Technology (2007)
44. Walsh, T.: Where are the really hard manipulation problems? The phase transition in manipulating the veto rule. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence, pp. 324–329. IJCAI (2009)

6 KONTROLLE IN MAXIMIN-WAHLEN

Wie das vorangegangene Kapitel 5, behandelt dieses Kapitel Kontrollprobleme bezüglich der Partitionierung der Kandidaten- und Wählermenge. Im Fokus der zugrunde liegenden Arbeit steht die Wahlregel Maximin. Die Arbeit wurde in den Proceedings zu ECAI 2020 [54] veröffentlicht:

C. Maushagen und J. Rothe. „The Last Voting Rule Is Home: Complexity of Control by Partition of Candidates or Voters in Maximin Elections“. In: *Proceedings of the 24th European Conference on Artificial Intelligence*. IOS Press, 2020, S. 163–170.

6.1 ZUSAMMENFASSUNG

In dieser Arbeit betrachten wir die Wahlregel Maximin und untersuchen diese hinsichtlich verschiedener Kontrollprobleme, welche von Bartholdi et al. [5] und Hemaspaandra et al. [38] eingeführt wurden. Dabei wird Maximin mit Hilfe des *Maximin-Scores* eines Kandidaten definiert. Dieser ergibt sich aus der Anzahl der Wähler, welche diesen Kandidaten gegenüber seinem stärksten Kontrahenten im paarweisen Vergleich bevorzugen. Die Kandidaten mit dem höchsten Maximin-Score gewinnen die Wahl. Kurz gesagt gewinnt der Kandidat, dessen schlechtester paarweiser Vergleich besser ist als der aller anderen Kandidaten.
Die so definierte Wahlregel wurde bereits von Faliszewski et al. [28] für diverse Kontrollprobleme untersucht. Weitere Kontrollprobleme zu Maximin wurden von der Autorin zusammen mit Rothe [50] hinsichtlich ihrer Komplexität klassifiziert. In dieser Arbeit untersuchen wir die damals noch offen gebliebenen Kontrollprobleme, welche die Partitionierung der Kandidaten- und Wählermenge betreffen. Dabei beweisen wir, dass alle untersuchten Kontrollprobleme für Maximin NP-vollständig sind.

6.2 EIGENER ANTEIL

Die Arbeit wurde zusammen mit Jörg Rothe geschrieben. Die technischen Teile der Arbeit sind mir zuzuschreiben.

ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200089

The Last Voting Rule Is Home: Complexity of Control by Partition of Candidates or Voters in Maximin Elections

Cynthia Maushagen[1] and **Jörg Rothe**[1]

Abstract. One of the key topics of computational social choice is electoral control, which models certain ways of how an election chair can seek to influence the outcome of elections via structural changes such as adding, deleting, or partitioning either candidates or voters. Faliszewski and Rothe [13] have surveyed the rich literature on control, giving an overview of previous results on the complexity of the associated problems for the most important voting rules. Among those, only a few results were known for two quite prominent voting rules: Borda Count and maximin voting (a.k.a. the Simpson–Kramer rule). Neveling and Rothe [26, 25] recently settled the remaining open cases for Borda. In this paper, we solve all remaining open cases for the complexity of control in maximin elections all of which concern control by partition of either candidates or voters.

1 Introduction

Thirty years ago, in a series of seminal papers, Bartholdi et al. gave birth to the field of computational social choice by introducing and studying problems associated with winner determination for Dodgson and Kemeny elections [3], manipulation of elections [2, 1], and electoral control [4] in terms of their computational complexity. Later on, Hemaspaandra et al. [19, 22] pinpointed the complexity of determining Dodgson and Kemeny winners exactly, Faliszewski et al. [10, 12] introduced and studied bribery in elections, Conitzer et al. [7] studied coalitional weighted manipulation for the most important voting rules and also its destructive variant (where the manipulators' goal is not to make their favorite candidate win but to prevent their most despised candidate's victory), and Hemaspaandra et al. [20] studied destructive control. Among these research lines, we will here focus on *electoral control*.

Driven by the many applications of collective decision making (e.g., by voting) in artificial intelligence—ranging from automated scheduling [17] over recommender systems [15] to collaborative filtering [29], from computational linguistics [27] to information extraction [34], and from planning [9] to meta-searching the internet [8]—computational social choice has turned into an established area that is now a key topic of the major AI conferences (see, e.g., two recent papers in the AAAI Senior Member Track [23, 32]). This success story has been comprehensibly told in the *Handbook of Computational Social Choice* [6] and other books [31]. In particular, Faliszewski and Rothe [13] surveyed the state of the art in control (and bribery), summarizing the common control scenarios and the related complexity results for the most important voting rules.

However, results for two quite prominent voting rules in their chapter were scarce: By 2016, not much was known about the control complexity for the Borda Count and maximin voting, which is also known as the Simpson–Kramer rule. Borda is perhaps the most important rule within the class of scoring protocols: In Borda, voters rank the m candidates, the ith candidate in each ranking scores $m-i$ points, and whoever has the most points wins. Maximin, on the other hand, is a rule that, like Condorcet or Copeland, is based on pairwise comparisons: The candidates' maximin scores result from their worst pairwise comparison against other candidates and all candidates with the largest maximin score win.

While Neveling and Rothe [26, 25] recently settled the remaining open cases for Borda, in this paper we solve all remaining open cases regarding the control complexity in maximin elections. With our results, the "last voting rule is home" in the sense that we now have an almost[2] complete picture of the control complexity of all voting rules considered by Faliszewski and Rothe in their chapter [13]. All our results are NP-hardness results, that is, we will show that maximin is resistant to the corresponding types of control.

All these open issues for maximin voting concern control by partition of either candidates or voters. Previous results for maximin voting on the complexity of control by adding or deleting either candidates or voters are due to Faliszewski et al. [11], and some cases of destructive control by partition of candidates are due to Maushagen and Rothe [24]. We settle the remaining cases for maximin: constructive control by partition of candidates and constructive and destructive control by partition of voters.[3] In particular, control by partition of voters is very interesting, as it is a simple model of *gerrymandering* and therefore quite well motivated for application in the real world. It is also noteworthy that resistance to partition of candidates or voters typically is shown via the technically most involved proofs. Further, we have tried to simplify and unify our proofs as much as possible: Our eight NP-hardness results are shown via essentially only two constructions.

This paper is organized as follows. In Section 2, we give some background on elections and introduce some technical definitions that are helpful for our proofs. Our results for constructive control by partition of candidates in maximin elections are presented in Section 3 and our results for constructive and destructive control by partition of voters in Section 4. We conclude in Section 5.

2 Preliminaries

An election is given by a pair (C, V) with C being a set of candidates and V a profile of the voters' preferences over C. Each preference is a linear order over C. Identifying voters with the votes they cast,

[1] Institut für Informatik, Heinrich-Heine-Universität Düsseldorf, Germany. E-mail: {cynthia.maushagen, rothe}@hhu.de

[2] For Schulze voting [33], it is still open how hard destructive control by adding and deleting candidates is.

[3] These control scenarios will be defined formally in Sections 3 and 4.

we will use the words *vote* and *voter* interchangeably. A vote of the form $b\ a\ c$ indicates that b is preferred to a and a to c.

Let $S \subseteq C$ be a subset of the candidates. When we write $\overrightarrow{S}$ in a vote, we mean a ranking of the candidates of S occurring in this vote in an arbitrary but fixed order; and when we write $\overleftarrow{S}$ in a vote, we mean their ranking in this vote in reverse order; and the order of the candidates from S does not matter in a vote when we simply write S in it. For example, letting $C = \{a, b, c, d\}$ and $S = \{b, c\}$ and assuming the lexicographic order of candidates, $d\ \overrightarrow{S}\ a$ means the vote $d\ b\ c\ a$; yet $d\ \overleftarrow{S}\ a$ means the vote $d\ c\ b\ a$; and $d\ S\ a$ could mean either of the votes $d\ b\ c\ a$ and $d\ c\ b\ a$.

To conveniently construct votes, for a set C of candidates and $c, d \in C$, let

$$W(c,d) = (c\ d\ \overrightarrow{C \setminus \{c,d\}},\ \overleftarrow{C \setminus \{c,d\}}\ c\ d)$$

be a pair of votes whose addition to a maximin election has the following effect on the scores of c and d: For such a pair, under maximin voting c gains two points in the head-to-head contest on d. For each other pair of candidates, they both gain one point in their head-to-head contest.

Some background from complexity theory is assumed, including standard notions such as the complexity classes P and NP, polynomial-time many-one-reducibility, and NP-hardness and NP-completeness. For more details, we refer to the textbooks by Garey and Johnson [14], Papadimitriou [28], and Rothe [30].

In our reductions for establishing NP-hardness, we will employ the following well-known NP-complete problems [14]. In ONE-IN-THREE-POSITIVE-3SAT, we are given a set X of boolean variables and a set $\mathcal{S}$ of clauses over X, each with exactly three unnegated literals, and we ask whether there is a truth assignment to the variables in X satisfying that in each clause of $\mathcal{S}$ exactly one literal is set to true. In EXACT-COVER-BY-3-SETS (X3C), we are given a set $B = \{b_1, \ldots, b_{3k}\}$ and a family $\mathcal{S} = \{S_1, \ldots, S_n\}$ of sets such that $S_i \subseteq B$ and $|S_i| = 3$ for all $S_i \in \mathcal{S}$, and the question is: Does there exist an exact cover of B, i.e., a subset $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| = k$ and $\bigcup_{S_i \in S'} S_i = B$?

3 Constructive Control by Partition of Candidates

In this section, we consider the standard scenarios of constructive control by partition of candidates and solve all remaining open cases regarding the complexity of candidate control in maximin elections. In Table 1, we give an overview of all previously known and new results on the complexity of candidate control for maximin. We start by defining in Section 3.1 the problems we consider and then prove our results in Section 3.2.

3.1 Problem Definitions and Overview of Results

We first consider constructive control by partition of candidates, as defined by Bartholdi et al. [4] for any given voting rule $\mathcal{E}$ (which here will always be maximin). In this scenario, the election (C, V) is held in two rounds and we assume that the chair has the power to subdivide the candidates into two groups, C_1 and C_2, and the winners of the first-round subelection (C_1, V) (where we tacitly assume that the votes in V are restricted to the candidates in C_1) run against the candidates in C_2, i.e., all members of C_2 get a bye to the final round.

We will adopt the so-called *unique-winner model* (as Bartholdi et al. [4] did), which means that for a control action to be successful, it is required that the distinguished candidate is the *only* winner. By contrast, in the *nonunique-winner model*, which also has been studied intensively for control problems [13, 5], it would be enough that the distinguished candidate is one among possibly several winners for a control action to be successful. Note that our results, even though expressed in the unique-winner model, hold also in the nonunique-winner model, as can be shown by slight modifications of our proofs.[4]

We will use the following tie-handling rules due to Hemaspaandra et al. [21]: According to *ties-promote* (TP) all winners of a first-round subelection will move forward to the final round and according to *ties-eliminate* (TE) only a unique first-round subelection winner moves forward to the final round (i.e., if there are two or more first-round subelection winners, they eliminate each other and no one moves to the final round from this first-round subelection).[5]

Now we are ready to define the first decision problem, $\mathcal{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-CANDIDATES-TP for voting rule $\mathcal{E}$, which given an election (C, V) and a distinguished candidate $p \in C$, asks whether we can partition C into C_1 and C_2 such that p is the unique $\mathcal{E}$ winner of the two-round election where the winners of the first-round subelection (C_1, V) run against all members of C_2 in a final round (with the votes from V in all subelections appropriately restricted to the participating candidates.

This problem name is abbreviated by $\mathcal{E}$-CCPC-TP. The related problem $\mathcal{E}$-CONSTRUCTIVE-CONTROL-BY-RUNOFF-PARTITION-OF-CANDIDATES-TP ($\mathcal{E}$-CCRPC-TP, for short; also due to Bartholdi et al. [4]) is defined similarly, except that now we have two first-round subelections, (C_1, V) and (C_2, V), and the winners of both proceed to the final runoff. With the other tie-handling rule, ties-eliminate, we receive the corresponding problems $\mathcal{E}$-CCPC-TE and $\mathcal{E}$-CCRPC-TE.

The destructive variants of these four problems, due to Hemaspaandra et al. [21], are defined analogously, except that the chair's goal now is to prevent the victory of the distinguished candidate. We abbreviate the corresponding problems by $\mathcal{E}$-DCPC-TP, $\mathcal{E}$-DCRPC-TP, $\mathcal{E}$-DCPC-TE, and $\mathcal{E}$-DCRPC-TE.[6]

Further, Bartholdi et al. [4] and Hemaspaandra et al. [21] introduced and studied for various voting rules the notions of constructive and destructive control by adding candidates (CCAC and DCAC), by adding an unlimited number of candidates (CCAUC and DCAUC), and by deleting candidates (CCDC and DCDC). As we do not study these control scenarios here, we refrain from defining them formally, instead referring to the work of Bartholdi et al. [4] and Hemaspaandra et al. [21] and in particular to the work of Faliszewski et al. [11] who obtained results for them in maximin elections. Table 1 gives an overview of all previous complexity results for candidate control in maximin elections (which are due to Faliszewski et al. [11] and Maushagen and Rothe [24]) as well as the new complexity results for candidate control in maximin elections established in this paper.

A voting rule $\mathcal{E}$ maps each election (C, V) to a subset of can-

[4] In fact, the constructions need not be changed, as in the yes-instances p will always win alone, whereas in the no-instances p will never even win. That is, the stronger condition of the unique-winner or the nonunique-winner model will always be satisfied. All that needs to be changed in the proofs for them to work also in the nonunique-winner model are minor modifications of the wording in the argumentation.

[5] Note that the unique-winner model better fits the TE rule and the nonunique-winner model better fits the TP rule.

[6] Hemaspaandra et al. [18] noted that, depending on whether we use TP or TE and on what winner model we choose (i.e., either the unique-winner or the nonunique-winner model), DCPC and DCRPC can be identical problems. Specifically, in the unique-winner model, we have DCRPC-TE = DCPC-TE, and in the nonunique-winner model, we have DCRPC-TE = DCPC-TE and DCRPC-TP = DCPC-TP.

C	CAUC D	C	CAC D	C	CDC D	C	CPC-TE D	C	CPC-TP D	C	CRPC-TE D	C	CRPC-TP D
V♡	V♡	R♡	V♡	V♡	V♡	**R**	V♠	**R**	V♠	**R**	V♠	**R**	V♠

Table 1. Overview of complexity results for candidate control in maximin elections. R means resistance and V means vulnerability. Results in boldface are established in this paper, and previous results are due to Faliszewski et al. [11] (marked by ♡) and due to Maushagen and Rothe [24] (marked by ♠).

didates, the *winners* of the election. We focus on the *maximin* voting rule (a.k.a. *Simpson–Kramer's rule*), which is based on pairwise comparisons. Given an election (C, V), for any two candidates $c, d \in C$, we denote the number of voters preferring c to d by $N_V(c, d)$. The *maximin score of candidate* c then is

$$\text{score}_{(C,V)}(c) = \min_{d \in C \setminus \{c\}} N_V(c, d) - N_V(d, c),$$

and whoever has the largest maximin score wins the election. In the following, we will omit the subscripts and simply write $\text{score}(c)$ and $N(c, d)$ if the meaning is clear from the context. A *Condorcet winner* is a candidate who wins each pairwise comparison; thus a Condorcet winner is always a maximin winner.

A voting rule $\mathcal{E}$ is said to be *immune to a control type* $\mathfrak{C}$ (such as constructive control by partition of candidates when ties promote) if it is never possible for the chair to reach her control goal; otherwise, it is said to be *susceptible to* $\mathfrak{C}$. Note that maximin is easily seen to be susceptible to every control type considered in this paper. If $\mathcal{E}$ is susceptible to $\mathfrak{C}$, we are interested in the computational complexity of the associated control problem (such as $\mathcal{E}$-CCPC-TP). We say that $\mathcal{E}$ is *vulnerable to* $\mathfrak{C}$ if $\mathcal{E}$ is susceptible to $\mathfrak{C}$ and the control problem corresponding to $\mathfrak{C}$ can be solved in polynomial time, and we say $\mathcal{E}$ is *resistant to* $\mathfrak{C}$ if $\mathcal{E}$ is susceptible to $\mathfrak{C}$ and the corresponding control problem is NP-hard.

3.2 Results and Proofs

While maximin is vulnerable to destructive control by partition and runoff partition of candidates in model TP and TE [24], we will now show that it is resistant to *constructive* control by partition and runoff partition of candidates with both tie-handling rules, TP and TE. For each of these four problems, we can use the same reduction.

Theorem 3.1. *For maximin elections, each of the problems* CCPC-TP*,* CCRPC-TP*,* CCPC-TE*, and* CCRPC-TE *is* NP*-complete.*

Proof. Membership of all four problems in NP is obvious. To show NP-hardness, we reduce from ONE-IN-THREE-POSITIVE-3SAT. Let $(X, \mathcal{S})$ be a ONE-IN-THREE-POSITIVE-3SAT instance with $X = \{x_1, \ldots, x_m\}$ and $\mathcal{S} = \{S_1, \ldots, S_n\}$, where $S_i = \{x_{i,1}, x_{i,2}, x_{i,3}\} \subseteq X$ and $|S_i| = 3$ for each $1 \leq j \leq n$. In the following, we use each $x \in X$ and each $S \in \mathcal{S}$ both as part of the given instance of ONE-IN-THREE-POSITIVE-3SAT and as the candidates of the election that is part of the constructed instance of any of the four control-by-partition problems. It will always be clear from the context what meaning is intended.

Specifically, from $(X, \mathcal{S})$ we construct an election (C, V) with the set $C = \{p, d, w\} \cup X \cup \mathcal{S} \cup R \cup T$ of candidates with $R = \{r_{i,j} \mid 1 \leq i \leq n, 1 \leq j \leq 3\}$ and $T = \{t_{i,j} \mid r_{i,j} \in R\}$.

The list V of votes is constructed as follows, where we write $[n]$ for the set $\{1, \ldots, n\}$ for $n \in \mathbb{N}$:

#	preference	for each
1	$W(r, p)$	$r \in R$
1	$W(t, p)$	$t \in T$
1	$W(w, S)$	$S \in \mathcal{S}$
1	$W(w, r)$	$r \in R$
2	$W(p, x)$	$x \in X$
2	$W(p, d)$	
4	$W(r, x)$	$r \in R, x \in X$
4	$W(x_{i,j}, t_{i,j})$	$i \in [n], j \in [3]$
4	$W(t_{i,j}, r_{i,j})$	$i \in [n], j \in [3]$
4	$W(x_{i,j}, t_{i,(j+1) \bmod 3})$	$i \in [n], j \in [3]$
$1 + j - i$	$W(S_i, S_j)$	$1 \leq i < j \leq n$
$n + 1$	$W(S, r)$	$S \in \mathcal{S}, r \in R$
$n + 1$	$W(S, d)$	
$n + 2$	$W(d, w)$	
$n + 3$	$W(w, x)$	$x \in X$
$n + 3$	$W(x, S)$	$S \in \mathcal{S}, x \in S$
$n + 3$	$W(w, t)$	$t \in T$
$n + 3$	$W(t, d)$	$t \in T$
$n + 3$	$W(r, d)$	$r \in R$
$n + 4$	$W(w, p)$	
$n + 4$	$W(S, p)$	$S \in \mathcal{S}$

This construction is sketched in Figures 1 and 2. In particular, Figure 2 shows the subgraph among the candidates from the sets X, R, T, and $S_i \in \mathcal{S}$ for fixed i, where a directed edge between two candidates, say pointing from a to b, means that a wins the pairwise comparison against b. The edges are weighted and their positive integer weights give the numbers of how often preference $W(a, b)$ occurs in the construction. Doubling these weights gives the surplus indicating how strongly a wins against b. If an edge starts from a gray rectangle, this means that *all* candidates from this set beat the candidates the edge points to, and similarly so the other way around: an edge pointing to a gray rectangle means that *all* candidates in this set are beaten by the candidate this edge originates from. In the graph shown in Figure 1, on the other hand, these subgraphs from Figure 2 are only roughly adumbrated (each framed by a dotted line).

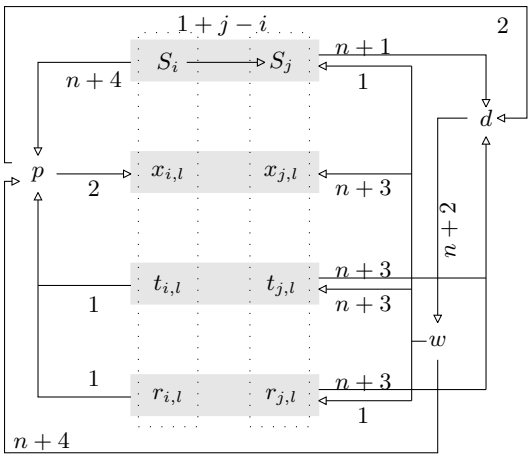

Figure 1. Construction in the proof of Theorem 3.1: Overview

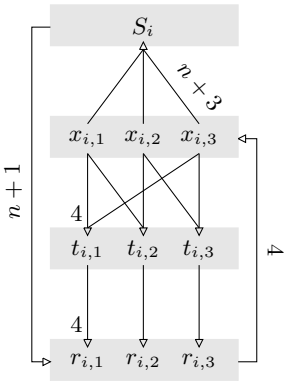

Figure 2. Construction in the proof of Theorem 3.1: subgraph for one S_i

Let $\Pi \in \{\text{CCRPC-TP}, \text{CCPC-TP}, \text{CCRPC-TE}, \text{CCPC-TE}\}$. We show: $(X, \mathcal{S})$ is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT if and only if (C, V, p) is a yes-instance of maximin-Π.

From left to right, let $(X, \mathcal{S})$ be a yes-instance of ONE-IN-THREE-POSITIVE-3SAT. Then there is a subset $U \subseteq X$ such that for each clause S_i we have $|U \cap S_i| = 1$. Partition the set C of candidates into C_1 and C_2 with

$$C_1 = \{d, w\} \cup \mathcal{S} \cup U \quad \text{and}$$
$$C_2 = \{p\} \cup R \cup T \cup X \setminus U.$$

Let us start with the case where we have two first-round subelections (i.e., with constructive control by runoff partition of candidates). In the subelection (C_1, V), each candidate $x \in U$ and each candidate $S \in \mathcal{S}$ has a score of $-2(n+3)$, w has a score of $-2(n+2)$, and d wins the subelection with a score of $-2(n+1)$. Since d is the unique winner of (C_1, V), d will move forward to the final round, regardless of the tie-breaking rule. In the subelection (C_2, V), the candidate p gets a score of -2, while each other candidate gets a score of -8. Since p is the unique winner of the subelection (C_2, V), p will move forward to the final round, regardless of the tie-breaking rule. In the final round, p faces only d and only p wins this head-on-head contest and thus the election.

Let us now consider the case of constructive control by partition of candidates where we have only one first-round subelection, (C_1, V), and all candidates from C_2 move directly forward to the final round. Compared with the subelection (C_2, V), the occurrence of d does not affect the score of any candidate. Since d has a score smaller than -4, p is again the only winner of the final election.

It follows that (C, V, p) is a yes-instance of maximin-Π for each $\Pi \in \{\text{CCPC-TP}, \text{CCRPC-TP}, \text{CCPC-TE}, \text{CCRPC-TE}\}$, completing the proof of the desired equivalence from left to right.

Conversely, from right to left, assuming that $(X, \mathcal{S})$ is a no-instance of ONE-IN-THREE-POSITIVE-3SAT, we will show that the constructed instance (C, V, p) is a no-instance of maximin-Π for each $\Pi \in \{\text{CCPC-TP}, \text{CCRPC-TP}, \text{CCPC-TE}, \text{CCRPC-TE}\}$ as well, i.e., we show that p cannot be made a unique winner of the two-stage election resulting from any possible partition of the candidate set (with or without runoff and for both tie-handling rules).

The worst pairwise comparison for p is between p and w as well as between p and each candidate $S \in \mathcal{S}$. Thus, if p were to face any of these candidates in a first-round subelection or in the final round, p would not win according to this partition of C. Therefore, we have to show that for each partition of candidates, where w and all $S \in \mathcal{S}$ participate in (C_1, V) and p is in (C_2, V), p does not win the election.

We consider all remaining partitions below.

Case 1: Let $C_1 = \{w\} \cup \mathcal{S} \cup T' \cup R' \cup U$ with $T' \subseteq T$, $R' \subseteq R$, and $U \subseteq X$. Candidate w wins each head-to-head contest, so w is the unique winner and can move forward to the final election. It follows that p can not win the election.

Case 2: Let $C_1 = \{d, w\} \cup \mathcal{S} \cup U$ with $U \subseteq X$. We consider two subcases.

Case 2.1: For each $S \in \mathcal{S}$, it is $S \cap U \neq \emptyset$. Since, we started with a no-instance of ONE-IN-THREE-POSITIVE-3SAT, there is at least one S_i such that $x_{i,j}, x_{i,j+1 \bmod 3} \in C_1$ with $1 \leq j \leq 3$. Each $S \in \mathcal{S}$ and each $x \in U$ has a score of $-2(n+3)$, w has a score of $-2(n+2)$ and d has a score of $-2(n+1)$. Therefore, d is the unique winner of the subelection and can move forward to the final round. Now, we have to destinguish between CCRPC and CCPC. We start with CCRPC. In (C_2, V), we have $C_2 = \{p\} \cup T \cup R \cup X \setminus U$. Candidate $t_{i,j+1 \bmod 3}$ has a score of 0, whereas p has a score of -2. It follows that p can not move forward to the final election. Let us now consider where we have only one subelection, CCPC. Since d is the unique winner of the first subelection, the final runoff is $(C_2 \cup \{d\}, V)$. Compared with the subelection (C_2, V), the occurrence of d does not affect the score of any candidate. It follows that p can not win the final election.

Case 2.2: It exists a $S \in \mathcal{S}$ with $S \cap U = \emptyset$. Let $\mathcal{S}' = \{S \mid S \cap U = \emptyset\} \subset \mathcal{S}$. Each $S_i \in \mathcal{S}'$ has a score of $-2i$, each $S_i \in \mathcal{S} \setminus \mathcal{S}'$ and each $x \in U$ has a score of $-2(n+3)$, d has a score of $-2(n+1)$, and w has a score of $-2(n+2)$. The candidate $S_i \in \mathcal{S}'$ with the lowest subscript wins the election.

Case 3: Let $C_1 = \{d, w\} \cup \mathcal{S} \cup T' \cup R'$ with $T' \subseteq T$, $R' \subseteq R$. Each S_i has a score of $-2i$ and each other candidate has a score lower than or equal to $-2(n+1)$. Therefore, S_1 is the unique winner of the subelection and can move forward to the final runoff, such that p can not win the election.

Case 4: Let $C_1 = \{d, w\} \cup \mathcal{S} \cup T' \cup R' \cup U$ with $T' \subseteq T$, $R' \subseteq R$. For $U = \emptyset$, we have Case 3 and for $T' \cup R' = \emptyset$, we have Case 2. We consider two subcases.

Case 4.1: It is $S \cap U \neq \emptyset$ for each $S \in \mathcal{S}$. Candidate w is the

unique winner with a score of $-2(n+2)$ while each other candidate has a score of $-2(n+3)$. Thus, p can not win the election.

Case 4.2: It exists $S \in \mathcal{S}$ such that $S \cap U = \emptyset$. Let $\mathcal{S}' = \{S \mid S \cap U = \emptyset\} \subset \mathcal{S}$. Each $S_i \in \mathcal{S}'$ has a score of $-2i$, w has a score of $-2(n+2)$ and each other candidate has a score of $-2(n+3)$. The candidate $S_i \in \mathcal{S}'$ with the lowest subscript is the unique winner of the subelection. It follows that p can not win the election.

It follows that (C, V, p) is a no-instance of maximin-Π for each $\Pi \in \{\text{CCRPC-TP}, \text{CCPC-TP}, \text{CCRPC-TE}, \text{CCPC-TE}\}$. ❑

4 Control by Partition of Voters

We now turn to constructive and destructive control by partition of voters, again with the tie-handling rules TP and TE, thus solving all remaining open cases for the complexity of voter control in maximin elections. In Table 2, we give an overview of all previously known and new results on the complexity of voter control for maximin.

We start by defining in Section 4.1 the problems we consider and will then prove our results in Section 4.2.

4.1 Problem Definitions and Overview of Results

These problems have also been introduced and studied by Bartholdi et al. [4] and Hemaspaandra et al. [21]. In control by partition of voters, the election (C, V) is again held in two rounds but now we assume that the chair has the power to subdivide the *voters* into two groups, V_1 and V_2, and the winners of the two first-round subelections (C, V_1) and (C, V_2) run against each other in a final round. This can be seen as a very simple model of *gerrymandering*.

Specifically, for any given voting rule $\mathcal{E}$ (which again will always be maximin), we give the formal definition of one of the decision problems we study in detail: In $\mathcal{E}$-CONSTRUCTIVE-CONTROL-BY-PARTITION-OF-VOTERS-TP, we are given an election (C, V) and a distinguished candidate $p \in C$, and we ask whether V can be partitioned into V_1 and V_2 such that p is the unique $\mathcal{E}$ winner of the two-round election where the winners of the two first-round subelections, (C, V_1) and (C, V_2), run against each other in a final round.

We abbreviate this problem by $\mathcal{E}$-CCPV-TP. The related problem for the other tie-handling rule, ties-eliminate, is $\mathcal{E}$-CCPV-TE, and the two destructive variants are $\mathcal{E}$-DCPV-TP and $\mathcal{E}$-DCPV-TE.

Bartholdi et al. [4] and Hemaspaandra et al. [21] also introduced and studied for various voting rules the notions of constructive and destructive control by adding voters (CCAV and DCAV) and by deleting voters (CCDV and DCDV). Again, we refrain from defining these formally, as we won't study these control scenarios here, and we instead refer to the work of Bartholdi et al. [4] and Hemaspaandra et al. [21] and in particular to that of Faliszewski et al. [11] who obtained results for them in maximin elections. Table 2 gives an overview of all previous complexity results for voters control in maximin elections (due to Faliszewski et al. [11]).

For notational convenience and to simplify our proofs, we use a slightly different scoring function in this section. Given an election (C, V) and a candidate $c \in C$, define $\text{Score}_{(C,V)}(c) = \min_{d \in C \setminus \{c\}} N_V(c, d)$, again omitting the subscript if (C, V) is clear from the context. Note that

$$\begin{aligned} N_V(c,d) - N_V(d,c) &= N_V(c,d) - (|V| - N_V(c,d)) \\ &= 2N_V(c,d) - |V|, \end{aligned}$$

so this is simply a linear shift compared with $\text{score}_{(C,V)}(c)$.

C CAV	D CAV	C CDV	D CDV	C CPV-TE	D CPV-TE	C CPV-TP	D CPV-TP
R♡	R♡	R♡	R♡	**R**	**R**	**R**	**R**

Table 2. Overview of complexity results for voter control in maximin elections. R means resistance and V means vulnerability. Results in boldface are established in this paper, and previous results are due to Faliszewski et al. [11] (marked by ♡).

4.2 Results and Proofs

We first consider control by partition of voters when ties promote.

Theorem 4.1. *maximin*-CCPV-TP *and maximin*-DCPV-TP *are* NP*-complete.*

Proof. Membership of these two problems in NP again is obvious. To show NP-hardness, we reduce from a variant of X3C. Let $(B, \mathcal{S})$ be an X3C instance, where $B = \{b_1, \ldots, b_{3k}\}$, $\mathcal{S} = \{S_1, \ldots, S_n\}$, and $S_i = \{b_{i,1}, b_{i,2}, b_{i,3}\}$ for each $S_i \in \mathcal{S}$. Without loss of generality, we may assume that $k > 6$. Furthermore, we may assume that each $b_j \in B$ is contained in exactly three sets $S_i \in \mathcal{S}$; thus $|B| = |\mathcal{S}| = n$. That X3C even with this restriction is still NP-hard was shown by Gonzalez [16].

From $(B, \mathcal{S})$, we construct an election with the set $C = \{p, d, w\} \cup B$ of candidates and with the distinguished candidate p for the constructive case and the distinguished candidate w for the destructive case. The list V of votes is constructed as follows:

#	preference	for each
$2k-1$	$w\ p\ B\ d$	
$4k-2$	$w\ d\ B\ p$	
1	$w\ B\ p\ d$	
1	$B \setminus S_i\ p\ d\ w\ S_i$	$S_i \in \mathcal{S}$
k	$B\ p\ w\ d$	
$2k$	$B\ d\ w\ p$	

In total, we have $12k-2$ votes. In these votes, the candidates from B are shifted cyclically such that, in particular:

- the first vote is $w\ p\ b_1\ b_2\ \cdots\ b_{3k-1}\ b_{3k}\ d$,
- the second vote is $w\ p\ b_{3k}\ b_1\ b_2\ \cdots\ b_{3k-1}\ d$, and
- the last vote is $b_4\ \cdots\ b_{3k}\ b_1\ b_2\ b_3\ d\ w\ p$.

However, the description of the construction is not finished yet; there still is a final twist to take notice of, regarding the fourth row of the construction, which gives one vote of the form $B \setminus S_i\ p\ d\ w\ S_i$ for each $S_i \in \mathcal{S}$. When constructing these $3k$ votes, let us first pretend that in this fourth row there were a vote of the form $B\ p\ d\ w$ (with the elements of B cyclically shifted as described above). But then we change each such vote $B\ p\ d\ w$ in the following way. For the ith vote $B\ p\ d\ w$, we shift $b_{i,1}$, $b_{i,2}$, and $b_{i,3}$ (i.e., the elements of S_i) from their current position to the end of the vote, while keeping the relative order among $b_{i,1}$, $b_{i,2}$, and $b_{i,3}$. For illustration, consider the following example. Suppose $S_1 = \{b_1, b_2, b_3\}$. After pretending that we have only votes of the form $B\ p\ d\ w$ instead of $B \setminus S_i\ p\ d\ w\ S_i$, we have a vote $b_3\ b_4\ \cdots\ b_{3k}\ b_1\ b_2\ p\ d\ w$ according to the cyclic shifts. This vote is then changed into: $b_4\ \cdots\ b_{3k}\ p\ d\ w\ b_3\ b_1\ b_2$.

Example 4.2. *As an illustration of the entire construction, consider the following example. Note that, for the sake of convenience and readability, we assume $k = 2$ in this example (even though we actually require $k > 6$ for the proof to work). Let $B = \{b_1, \ldots, b_6\}$ and $\mathcal{S} = \{S_1, \ldots, S_6\}$ with $S_1 = \{b_1, b_2, b_3\}$, $S_2 = \{b_4, b_5, b_6\}$, $S_3 = \{b_2, b_3, b_6\}$, $S_4 = \{b_2, b_4, b_5\}$, $S_5 = \{b_1, b_3, b_4\}$, and $S_6 = \{b_1, b_5, b_6\}$. According to our construction, we first get the following votes.*

From the first row of our vote list, we obtain three votes:

$$w\ p\ b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ d,$$
$$w\ p\ b_6\ b_1\ b_2\ b_3\ b_4\ b_5\ d,$$
$$w\ p\ b_5\ b_6\ b_1\ b_2\ b_3\ b_4\ d.$$

From the second row of our vote list, we obtain six votes:

$$w\ d\ b_4\ b_5\ b_6\ b_1\ b_2\ b_3\ p,$$
$$w\ d\ b_3\ b_4\ b_5\ b_6\ b_1\ b_2\ p,$$
$$w\ d\ b_2\ b_3\ b_4\ b_5\ b_6\ b_1\ p,$$
$$w\ d\ b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ p,$$
$$w\ d\ b_6\ b_1\ b_2\ b_3\ b_4\ b_5\ p,$$
$$w\ d\ b_5\ b_6\ b_1\ b_2\ b_3\ b_4\ p.$$

From the third row of our vote list, we obtain one vote:

$$w\ b_4\ b_5\ b_6\ b_1\ b_2\ b_3\ p\ d.$$

Note that the cyclic shifts of the elements of B spread across the single lines of our vote list. Now, from the fourth row of our vote list, we obtain six further votes, and now the cyclic shifts of the elements of B are slightly tampered with as we have explained above. Pretending we had the vote $B\ p\ d\ w$ in the fourth row of our vote list, then the first vote would be $b_3\ b_4\ b_5\ b_6\ b_1\ b_2\ p\ d\ w$. According to $B \setminus S_1\ p\ d\ w\ S_1$ with $S_1 = \{b_1, b_2, b_3\}$, this gives the vote: $b_4\ b_5\ b_6\ p\ d\ w\ b_3\ b_1\ b_2$. That is, the six votes corresponding to the fourth row are changed from

$$b_3\ b_4\ b_5\ b_6\ b_1\ b_2\ p\ d\ w,$$
$$b_2\ b_3\ b_4\ b_5\ b_6\ b_1\ p\ d\ w,$$
$$b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ p\ d\ w,$$
$$b_6\ b_1\ b_2\ b_3\ b_4\ b_5\ p\ d\ w,$$
$$b_5\ b_6\ b_1\ b_2\ b_3\ b_4\ p\ d\ w,$$
$$b_4\ b_5\ b_6\ b_1\ b_2\ b_3\ p\ d\ w$$

to the six votes where the three candidates corresponding to the $S_i \in \mathcal{S}$ are moved to the end of these six votes, respecting their relative order:

$$b_4\ b_5\ b_6\ p\ d\ w\ b_3\ b_1\ b_2,$$
$$b_2\ b_3\ b_1\ p\ d\ w\ b_4\ b_5\ b_6,$$
$$b_1\ b_4\ b_5\ p\ d\ w\ b_2\ b_3\ b_6,$$
$$b_6\ b_1\ b_3\ p\ d\ w\ b_2\ b_4\ b_5,$$
$$b_5\ b_6\ b_2\ p\ d\ w\ b_1\ b_3\ b_4,$$
$$b_4\ b_2\ b_3\ p\ d\ w\ b_5\ b_6\ b_1.$$

Next, from the fifth row of our vote list, we obtain two votes:

$$b_3\ b_4\ b_5\ b_6\ b_1\ b_2\ p\ w\ d,$$
$$b_2\ b_3\ b_4\ b_5\ b_6\ b_6\ p\ w\ d.$$

Finally, from the sixth row of our vote list, we obtain four votes:

$$b_1\ b_2\ b_3\ b_4\ b_5\ b_6\ d\ w\ p,$$
$$b_6\ b_1\ b_2\ b_3\ b_4\ b_5\ d\ w\ p.$$
$$b_5\ b_6\ b_1\ b_2\ b_3\ b_4\ d\ w\ p,$$
$$b_4\ b_5\ b_6\ b_1\ b_2\ b_3\ d\ w\ p.$$

The reduction can obviously be computed in polynomial time. Due to the cyclic shifts, we have $N(b_1, b_{3k}) \leq 7$ and $N(b_i, b_{i-1}) \leq 7$ for each i, $2 \leq i \leq 3k$. Therefore, $\mathrm{Score}(b) \leq 7$ for each $b \in B$. For the remaining candidates, we have the following pairwise comparisons:

$N(\downarrow, \rightarrow)$	p	w	d	$b \in B$
p	–	$4k$	$6k$	$2k+2$
w	$8k-2$	–	$7k-2$	$6k+1$
d	$6k-2$	$5k$	–	$4k+1$

We notice that w is the unique maximin winner and also a Condorcet winner of the election.

We claim that $(B, \mathcal{S})$ is in X3C if and only if (C, V, p) is a yes-instance of maximin-CCPV-TP (respectively, (C, V, w) is a yes-instance of maximin-DCPV-TP).

From left to right, let $(B, \mathcal{S})$ be a yes-instance of X3C. Then there is a subset $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| = k$ and $\bigcup_{S_i \in \mathcal{S}'} S_i = B$. Partition V into V_1 and V_2, where V_1 contains all votes of the form $w\ p\ B\ d$, as well as $B\ p\ w\ d$ and the k votes $B \setminus S_i\ p\ d\ w\ S_i$ and $S_i \in \mathcal{S}'$. The remaining votes are in V_2. In (C, V_1), we have:

$N(\downarrow, \rightarrow)$	p	w	d	$b \in B$
p	–	$2k$	$4k-1$	$2k$
w	$2k-1$	–	$3k-1$	$2k$
d	0	k	–	1

So p is the sole winner of the first first-round subelection and moves forward to the final round. In (C, V_2), we have:

$N(\downarrow, \rightarrow)$	p	w	d	$b \in B$
p	–	$2k$	$2k+1$	2
w	$6k-1$	–	$4k-1$	$4k+1$
d	$6k-2$	$4k$	–	$4k$

The candidate d has with $4k$ the highest score and is the only winner of the second first-round subelection.

The final runoff thus is $(\{p, d\}, V)$. We have $6k$ of $12k-2$ votes who prefer p to d such that p is the unique winner of the runoff. It follows that (C, V, p) (respectively, (C, V, w)) is a yes-instance of maximin-CCPV-TP (respectively, maximin-DCPV-TP), as desired.

From right to left, assuming that $(B, \mathcal{S})$ is a no-instance of X3C, we will show that for each partition of the voters, w is always the unique winner of the election.

A Condorcet winner in an election (C, V) remains a Condorcet winner for each subset $C' \subseteq C$.[7] Since w is a Condorcet winner in (C, V), it is necessary that w cannot move forward to the final round, or else w's overall victory cannot be prevented.

We have $6k-2$ out of $12k-2$ votes who prefer w the most. In the following, we denote the number of voters with a vote of the form $w\ \cdots$ in the subelection (C, V_i) with ℓ_i for $i \in \{1, 2\}$ when V is partitioned into V_1 and V_2. Note that $N_{V_i}(w, c) \geq \ell_i$ for each $c \in C \setminus \{w\}$. To prevent the victory of w in (C, V_i), it is necessary that we have at least $\ell_i + 1$ votes of the form $b\ \cdots$ in V_i.

Suppose that a candidate $b \in B$ wins a subelection, say, without loss of generality, (C, V_1). Since $\mathrm{Score}(b) \leq 7$, there are at most 13 voters in V_1.

It follows that in (C, V_2), w receives at least $6k + 1 - 13 = 6k - 12$ points, while $\mathrm{Score}(p) \leq 2k + 2$, $\mathrm{Score}(d) \leq 4k + 1$, and $\mathrm{Score}(b) \leq 7$ for each $b \in B$, so w is a winner of the subelection and therefore the unique winner of the whole election.

The only possibility to prevent w's victory is that p and d are unique winners of the two first-round subelections. Without loss of

[7] This is due to Condorcet voting satisfying the so-called weak axiom of revealed preferences (see, e.g., [4, 21]).

generality, we assume that p wins in (C, V_1) and d wins in (C, V_2). Since $\mathrm{Score}_{(C,V_1)}(w) = \ell_1$, it is necessary that $N_{V_1}(p, w) = \ell_1 + 1$ and thus all votes of the form $B\ d\ w\ p$ have to be in V_2. The other way around, it is necessary that all votes of the form $B\ p\ w\ d$ are in V_1 such that $N_{V_2}(d, w) = \ell_2 + 1 > \ell_2 = \mathrm{Score}(w)$.

We have the following votes in V_1:

#	preference
x_1	$w\ p\ B\ d$
x_2	$w\ d\ B\ p$
x_3	$w\ B\ p\ d$
x_4	$B \setminus S\ p\ d\ w\ S$
k	$B\ p\ w\ d$

with $0 \le x_1 \le 2k-1, 0 \le x_2 \le 4k-2, x_3 \in \{0, 1\}, 0 \le x_4 \le 3k$ and the restriction that $x_1 + x_2 + x_3 + 1 = x_4 + k$. Let $\varepsilon_b = |\{S \in \mathcal{S} \,|\, b \in S \text{ and } B \setminus S\ p\ d\ w\ S \in V_1\}|$ and $\varepsilon = \min\{\varepsilon_b \,|\, b \in B\}$. Since each b is in exactly three $S \in \mathcal{S}$, we have $0 \le \varepsilon_b \le 3$. Candidate w has a score of $x_1 + x_2 + x_3$. For p, we have $N(p, w) = x_4 + k$ and $N(p, b) = x_1 + \varepsilon_b$ for each $b \in B$. We distinguish two cases for the score of p. In both cases, we will see that that $\mathrm{Score}(p) > \mathrm{Score}(w)$ is not possible unless violating one of the conditions above.

Case 1: We have $x_4 + k < x_1 + \varepsilon$ and thus $\mathrm{Score}(p) = x_4 + k$. Since $x_4 + k = x_1 + x_2 + x_3 + 1$, we have $x_2 + x_3 + 1 < \varepsilon$. For $\varepsilon > 1$, it is necessary that $x_4 \ge 2k + 1$, since we started with a no-instance. It follows that $x_4 + k \ge 3k + 1$ and $x_1 + x_2 + x_3 + 1 \le 2k + 1$, so $x_1 + x_2 + x_3 + 1 < x_4 + k$.

Case 2: We have $x_1 + \varepsilon \le x_4 + k$ and thus $\mathrm{Score}(p) = x_1 + \varepsilon$. To prevent the victory of w, it is necessary that $x_1 + \varepsilon > x_1 + x_2 + x_3$, thus $x_2 + x_3 \in \{0, 1, 2\}$. For $x_2 + x_3 \ge 1$, we receive $\varepsilon \ge 2$ and thus $x_4 \ge 2k + 1$. It follows that $x_4 + k \ge 3k + 1$ while $x_1 + x_2 + x_3 + 1 \le 2k - 1 + 2 + 1 = 2k + 2$. For $x_2 + x_3 = 0$, we have $\varepsilon \ge 1$ and $x_4 \ge k + 1$. Therefore, $x_1 + x_2 + x_3 + 1 \le 2k$ and $x_4 + k \ge 2k + 1$.

This completes the case distinction. We have shown that, regardless of how the voters are partitioned, w is always the unique winner. It follows that (C, V, p) (respectively, (C, V, w)) is a no-instance of maximin-CCPV-TP (respectively, maximin-DCPV-TP). ❑

Next, we turn to control by partition of voters when ties eliminate. Since the proofs of Theorems 4.1 and 4.3 are pretty similar and due to space limitations, we will only sketch the latter one.

Theorem 4.3. *maximin*-CCPV-TE *and maximin*-DCPV-TE *are* NP*-complete.*

Proof. Membership of these two problems in NP once more is obvious. To show NP-hardness, we slightly modify the reduction from X3C that was given in the proof of Theorem 4.1: From our given X3C instance $(B, \mathcal{S})$, we construct an election (C, V) as in that proof, except that V now contains an additional vote of the form $w\ B\ p\ d$. The pairwise comparison between the candidates is now:

$N(\downarrow, \rightarrow)$	p	w	d	$b \in B$
p	–	$4k$	$6k + 1$	$2k + 2$
w	$8k - 1$	–	$7k - 1$	$6k + 2$
d	$6k - 2$	$5k$	–	$4k + 1$

We claim that $(B, \mathcal{S})$ is in X3C if and only if (C, V, p) is a yes-instance of maximin-CCPV-TE (respectively, (C, V, w) is a yes-instance of maximin-DCPV-TE).

From left to right, let $(B, \mathcal{S})$ be a yes-instance of X3C. Let $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| = k$ and $\bigcup_{S \in \mathcal{S}'} S = B$. Partition V into V_1 and V_2 such that V_1 contains the following votes:

#	preference	for each
$2k - 1$	$w\ p\ B\ d$	
1	$B \setminus S_i\ p\ d\ w\ S_i$	$S_i \in \mathcal{S}'$
k	$B\ p\ w\ d$	

and V_2 contains all the remaining votes. We claim that p will be made the unique winner by this partition. In the first subelection, (C, V_1), we have $\mathrm{Score}(w) = N(w, p) = 2k-1, \mathrm{Score}(p) = N(p, b) = 2k$, $\mathrm{Score}(d) = N(d, p) = 0$, and $\mathrm{Score}(b) \le 7$. Therefore, candidate p proceeds to the final round from this subelection. In the second subelection, (C, V_2), we have $\mathrm{Score}(w) = N(w, d) = 4k$, $\mathrm{Score}(d) = N(d, b) = 4k$, $\mathrm{Score}(p) = N(p, b) = 2$, and $\mathrm{Score}(b) \le 7$. Thus d and w both win this subelection. The tie-handling rule TE blocks them both, so no one moves forward from this subelection, and p wins the final round. It follows that p is the only candidate in the final runoff and therefore the sole winner. It follows that (C, V, p) (respectively, (C, V, w)) is a yes-instance of maximin-CCPV-TE (respectively, maximin-DCPV-TE).

The direction from right to left can be shown similarly as in the proof of Theorem 4.1 and is, again, only sketched here due to space constraints. Candidate w not reaching the final runoff is equivalent to w losing in one subelection with a point difference of one to the winner and both w and another candidate winning in the other subelection so that the tie-handling rule prevents that w moves forward to the final round. It is sufficient to consider what happens if p and w win a subelection. Let $c \in \{p, d\}$ and c' be the other candidate. If c is the unique winner of the first subelection and c' is a winner of the other subelection, it follows that each vote $B\ c'\ w\ c$ is in V_2 and each vote $B\ c\ w\ c'$ is in V_1. In the following, we use ε like in the proof of Theorem 4.1. We again distinguish two cases.

Case 1: p is the unique winner in (C, V_1). In V_1 we have the following votes:

#	preference
x_1	$w\ p\ B\ d$
x_2	$w\ d\ B\ p$
x_3	$w\ B\ p\ d$
x_4	$B \setminus S_i\ p\ d\ w\ S_i$
k	$B\ p\ w\ d$

We have the restriction $x_1 + x_2 + x_3 + 1 = x_4 + k$. To ensure that p is the unique winner, it is necessary that $\mathrm{score}(w) = x_1 + x_2 + x_3 < \mathrm{score}(p) \le N(p, b) = x_1 + \varepsilon$. It follows that $x_2 + x_3 < \varepsilon$ and $\varepsilon \in \{1, 2, 3\}$. For $\varepsilon = 1$, we have $x_4 \ge k + 1$ and thus $x_4 + k \ge 2k + 1$, which is a contradiction since $x_1 \le 2k - 1$. For $\varepsilon \in \{2, 3\}$, we have $x_4 \ge 2k + 1$ and thus $x_1 + x_2 + x_3 + 1 \le x_1 + 3$ and $x_4 + 2k \ge 3k + 1$. It follows that $x_1 \ge 3k - 2$, which again is a contradiction.

Case 2: d is the unique winner in (C, V_1). The list of votes is as in Case 1, except that we have $2k$ votes $B\ d\ w\ p$ instead of the k votes $B\ p\ w\ d$. We have the restriction $x_1 + x_2 + x_3 + 1 = x_4 + 2k$. To ensure that d is the unique winner, it is necessary that $\mathrm{score}(w) = x_1 + x_2 + x_3 < \mathrm{score}(d) \le N(d, b) = x_2 + \varepsilon$. It follows that

$x_1 + x_3 < \varepsilon$. For $\varepsilon = 1$, we have $x_4 \geq k + 1$ and $x_1 = x_3 = 0$. In the other subelection, (C, V_2), we have $\text{score}(w) = 6k - 1 - x_2$ and $N(p, b) \leq 2k$. If p were winning subelection (C, V_2), we would have that $x_2 \geq 4k - 1$, which is a contradiction, though. For $\varepsilon = 2wehavex_4 \geq 2k + 1$, and for $\varepsilon = 3$ we have $x_4 = 3k$. In both cases, we receive a contradiction, since our restrictions cause that $x_2 \geq 4k - 1$ and $x_2 \geq 5k - 3$, respectively.

It follows that (C, V, p) (respectively, (C, V, w)) is a no-instance of maximin-CCPV-TE (respectively, maximin-DCPV-TE). ❑

5 Conclusions

We have completed the picture regarding the control complexity of maximin voting by solving the remaining eight open problems related to control by partition of candidates and voters for this rule. Thus, with the only exception of Schulze voting for which the complexity of destructive control by adding and deleting candidates is still open, the control complexity for (almost) all voting rules mentioned by Faliszewski and Rothe [13] in their chapter on control and bribery in the *Handbook of Computational Social Choice* have been settled now. Chapter (almost) closed.

Acknowledgements: We thank the reviewers for helpful comments. This work was supported by DFG grant RO 1202/14-2.

References

[1] J. Bartholdi III and J. Orlin, 'Single transferable vote resists strategic voting', *Social Choice and Welfare*, **8**(4), 341–354, (1991).

[2] J. Bartholdi III, C. Tovey, and M. Trick, 'The computational difficulty of manipulating an election', *Social Choice and Welfare*, **6**(3), 227–241, (1989).

[3] J. Bartholdi III, C. Tovey, and M. Trick, 'Voting schemes for which it can be difficult to tell who won the election', *Social Choice and Welfare*, **6**(2), 157–165, (1989).

[4] J. Bartholdi III, C. Tovey, and M. Trick, 'How hard is it to control an election?', *Mathematical and Computer Modelling*, **16**(8/9), 27–40, (1992).

[5] D. Baumeister and J. Rothe, 'Preference aggregation by voting', in *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, ed., J. Rothe, Springer Texts in Business and Economics, chapter 4, 197–325, Springer-Verlag, (2015).

[6] *Handbook of Computational Social Choice*, eds., F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, Cambridge University Press, 2016.

[7] V. Conitzer, T. Sandholm, and J. Lang, 'When are elections with few candidates hard to manipulate?', *Journal of the ACM*, **54**(3), Article 14, (2007).

[8] C. Dwork, R. Kumar, M. Naor, and D. Sivakumar, 'Rank aggregation methods for the web', in *Proceedings of the 10th International World Wide Web Conference*, pp. 613–622. ACM Press, (2001).

[9] E. Ephrati and J. Rosenschein, 'A heuristic technique for multi-agent planning', *Annals of Mathematics and Artificial Intelligence*, **20**(1–4), 13–67, (1997).

[10] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra, 'How hard is bribery in elections?', *Journal of Artificial Intelligence Research*, **35**, 485–532, (2009).

[11] P. Faliszewski, E. Hemaspaandra, and L. Hemaspaandra, 'Multimode control attacks on elections', *Journal of Artificial Intelligence Research*, **40**, 305–351, (2011).

[12] P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra, and J. Rothe, 'Llull and Copeland voting computationally resist bribery and constructive control', *Journal of Artificial Intelligence Research*, **35**, 275–341, (2009).

[13] P. Faliszewski and J. Rothe, 'Control and bribery in voting', in *Handbook of Computational Social Choice*, eds., F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. Procaccia, chapter 7, 146–168, Cambridge University Press, (2016).

[14] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, 1979.

[15] S. Ghosh, M. Mundhe, K. Hernandez, and S. Sen, 'Voting for movies: The anatomy of recommender systems', in *Proceedings of the 3rd Annual Conference on Autonomous Agents*, pp. 434–435. ACM Press, (1999).

[16] T. Gonzalez, 'Clustering to minimize the maximum intercluster distance', *Theoretical Computer Science*, **38**, 293–306, (1985).

[17] T. Haynes, S. Sen, N. Arora, and R. Nadella, 'An automated meeting scheduling system that utilizes user preferences', in *Proceedings of the 1st International Conference on Autonomous Agents*, pp. 308–315. ACM Press, (1997).

[18] E. Hemaspaandra, L. Hemaspaandra, and C. Menton, 'Search versus decision for election manipulation problems', in *Proceedings of the 30th Annual Symposium on Theoretical Aspects of Computer Science*, volume 20 of *LIPIcs*, pp. 377–388. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, (February 2013).

[19] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe, 'Exact analysis of Dodgson elections: Lewis Carroll's 1876 voting system is complete for parallel access to NP', *Journal of the ACM*, **44**(6), 806–825, (1997).

[20] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe, 'Anyone but him: The complexity of precluding an alternative', in *Proceedings of the 20th National Conference on Artificial Intelligence*, pp. 95–101. AAAI Press, (2005).

[21] E. Hemaspaandra, L. Hemaspaandra, and J. Rothe, 'Anyone but him: The complexity of precluding an alternative', *Artificial Intelligence*, **171**(5–6), 255–285, (2007).

[22] E. Hemaspaandra, H. Spakowski, and J. Vogel, 'The complexity of Kemeny elections', *Theoretical Computer Science*, **349**(3), 382–391, (2005).

[23] L. Hemaspaandra, 'Computational social choice and computational complexity: BFFs?', in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 7971–7977. AAAI Press, (February 2018).

[24] C. Maushagen and J. Rothe, 'Complexity of control by partitioning veto and maximin elections and of control by adding candidates to plurality elections', in *Proceedings of the 22nd European Conference on Artificial Intelligence*, pp. 277–285. IOS Press, (August/September 2016).

[25] M. Neveling and J. Rothe, 'Closing the gap of control complexity in Borda elections: Solving ten open cases', in *Proceedings of the 18th Italian Conference on Theoretical Computer Science*, volume 1949, pp. 138–149. CEUR-WS.org, (September 2017).

[26] M. Neveling and J. Rothe, 'Solving seven open problems of offline and online control in Borda elections', in *Proceedings of the 31st AAAI Conference on Artificial Intelligence*, pp. 3029–3035. AAAI Press, (February 2017).

[27] K. Oflazer and G. Tür, 'Morphological disambiguation by voting constraints', in *Proceedings of the 8th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 222–229. ACL/Morgan Kaufmann, (1997).

[28] C. Papadimitriou, *Computational Complexity*, Addison-Wesley, second edn., 1995.

[29] D. Pennock, E. Horvitz, and C. Giles, 'Social choice theory and recommender systems: Analysis of the axiomatic foundations of collaborative filtering', in *Proceedings of the 17th National Conference on Artificial Intelligence*, pp. 729–734. AAAI Press, (July/August 2000).

[30] J. Rothe, *Complexity Theory and Cryptology. An Introduction to Cryptocomplexity*, EATCS Texts in Theoretical Computer Science, Springer-Verlag, 2005.

[31] *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, ed., J. Rothe, Springer Texts in Business and Economics, Springer-Verlag, 2015.

[32] J. Rothe, 'Borda count in collective decision making: A summary of recent results', in *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pp. 9830–9836. AAAI Press, (January/February 2019).

[33] M. Schulze, 'A new monotonic, clone-independent, reversal symmetric, and Condorcet-consistent single-winner election method', *Social Choice and Welfare*, **36**(2), 267–303, (2011).

[34] G. Sigletos, G. Paliouras, C. Spyropoulos, and M. Hatzopoulos, 'Combining information extraction systems using voting and stacked generalization', *Journal of Machine Learning Research*, **6**, 1751–1782, (2005).

6.3 KORRIGENDUM

Im Beweis von Theorem 4.1 zur NP-Härte von Maximin-CCPV-TP und Maximin-DCPV-TP ist ein Argumentationsfehler aufgetreten. Dieser befindet sich im zweiten Teil des Beweises, wo angenommen wird, dass eine Nein-Instanz zu X3C vorliegt.
Es wird im Beweis behauptet, dass bei einer Vorwahl in der ein Kandidat $b \in B$ gewinnt, höchstens 13 Wähler teilnehmen können. Bei einer Vorwahl an der nun allerdings alle $2k$ Wähler mit einer Präferenz der Form $Bdwp$ teilnehmen, gewinnt offensichtlich ein Kandidat $b \in B$ und für $k \geq 7$ haben wir stets mehr als 13 Wähler.
Stattdessen muss im Beweis wie folgt argumentiert werden:
Angenommen ein Kandidat $b \in B$ gewinnt eine Vorwahl, oBdA können wir annehmen, dass dies die Vorwahl (C, V_1) ist. Da der Punktestand eines Kandidaten $b \in B$ höchstens 7 sein kann, können höchstens 7 Wähler mit einer Stimme der Form $w \ldots$ an dieser Vorwahl teilnehmen. Damit nehmen also mindestens $6k - 9$ Wähler mit einer Stimme der Form $w \ldots$ an der zweiten Vorwahl (C, V_2) teil, wodurch der Punktestand von w mindestens $6k - 9$ beträgt. Die Kandidaten $b \in B$ können weiterhin höchstens einen Punktestand von 7 erreichen, p höchstens $2k + 1$ und d höchstens $4k + 1$. Der Kandidat w ist also auch nach der neuen Argumentation der alleinige Sieger der zweiten Vorwahl und der gesamten Wahl.

7

SHIFT-BRIBERY

In diesem Kapitel wird die Komplexität für das Bestechungsproblem Shift-Bribery hinsichtlich verschiedener iterativer Wahlregeln untersucht. Die Arbeit wurde bei der Fachzeitschrift [48] eingereicht:

C. Maushagen, M. Neveling, J. Rothe und A. Selker. „Complexity of Shift Bribery in Iterative Elections". Manuscript submitted to *Journal of Autonomous Agents and Multi-Agent Systems*.

Teile dieser Arbeit wurden bereits in einer früheren Arbeit mit den selben Autoren bei AAMAS 2018 [49] veröffentlicht.

7.1 ZUSAMMENFASSUNG

In diesem Kapitel untersuchen wir das Bestechungsproblem Shift-Bribery, welches von Elkind et al. [23] eingeführt wurde. Alle untersuchten Wahlregeln verbindet, dass diese rundenbasiert sind und in jeder Runde eine der drei Wahlregeln Plurality, Veto oder Borda genutzt wird, um einen Punktestand zu ermitteln. Die Unterschiede in den Wahlregeln ergeben sich aus der Festlegung, wann ein Kandidat an der nächsten Runde teilnehmen darf. Die von uns untersuchten Wahlregeln lassen sich in vier Gruppen aufteilen. Zur ersten Gruppe gehören die Wahlregeln *Hare*, *Coombs* und *Baldwin*, welche Plurality, Veto bzw. Borda zur Berechnung des Scores nutzen. Bei jeder dieser Wahlregeln scheiden die Kandidaten mit dem niedrigsten Punktestand solange aus, bis alle übrig gebliebenen Kandidaten einen Gleichstand erzielen. Die zweite Gruppe umfasst nur die Wahlregel *Nanson*, welche wie Baldwin in jeder Runde Borda verwendet um einen Punktestand zu ermitteln. Allerdings dürfen hier nur solche Kandidaten an der nächsten Runde teilnehmen, deren erzielter Borda-Score größer ist als der durchschnittliche Borda-Score dieser Runde. Die Wahlregeln *Iterated Plurality* und *Iterated Veto* bilden die dritte Gruppe von uns untersuchten Wahlregeln. Hier können Kandidaten nur an der nächsten Runde teilnehmen, wenn sie zu den Plurality- bzw. Veto-Gewinnern gehören. Bei der letzten Gruppe von Wahlregeln, welche *Plurality with runoff* und *Veto with runoff* umfasst, gibt es im Gegensatz zu den zuvor definierten Wahlregeln immer nur genau zwei Runden. Hier kommen die Kandidaten mit höchsten Plurality- bzw. Veto-Score in die finale zweite Runde und der Plurality- bzw. Veto-Sieger der zweiten Runde gewinnt die gesamte Wahl. Gibt es in der ersten Runde nur einen Plurality- bzw. Veto-Sieger so kommen alle Kandidaten mit den zweitmeisten Punkten auch ins Finale. In der Literatur gibt es

zu diesen Wahlregeln verschiedene Varianten, die sich dahingehend unterscheiden, dass Gleichstandsbrechungsregeln genutzt werden, damit in jeder Runde genau ein Kandidat ausscheidet. Die von uns gewählten Definitionen orientieren sich an dem Buch von Taylor [72]. Wir zeigen, dass das Problem SHIFT-BRIBERY für alle untersuchten Wahlregeln NP-vollständig ist.

7.2 EIGENER ANTEIL

Die Arbeit wurde zu gleichen Teilen zusammen mit meinen Koautoren geschrieben. Die Beweise zu den Theoremen 1, 3, 4, 9 und 10 sind mir zuzuschreiben.

Autonomous Agents and Multi-Agent Systems manuscript No.
(will be inserted by the editor)

Complexity of Shift Bribery in Iterative Elections

Cynthia Maushagen · **Marc Neveling** ·
Jörg Rothe · **Ann-Kathrin Selker**

Received: date / Accepted: date

Abstract In iterative voting systems, candidates are eliminated in consecutive rounds until either a fixed number of rounds is reached or the set of remaining candidates does not change anymore. We focus on iterative voting systems based on the positional scoring rules plurality, veto, and Borda and study their resistance against shift bribery attacks due to Elkind et al. [16] and Kaczmarczyk and Faliszewski [25]. In constructive shift bribery [16], an attacker seeks to make a designated candidate win the election by bribing voters to shift this candidate in their preferences; in destructive shift bribery [25], the briber's goal is to prevent this candidate's victory. We show that many iterative voting systems are resistant to these types of attack, i.e., the corresponding decision problems are NP-hard. These iterative voting systems include iterated plurality as well as the voting rules due to Hare (see, e.g., Taylor [38]), Coombs (see, e.g., [28]), Baldwin [1], and Nanson [30]; variants of Hare voting are also known as single transferable vote, instant-runoff voting, and alternative vote.

Keywords computational social choice · iterative voting · bribery · shift bribery · Hare election · Coombs election · Baldwin election · Nanson election

1 Introduction

One of the main themes in computational social choice [7,35] is to study the complexity of manipulative attacks on voting systems, in the hope that proving computational hardness of such attacks may provide some sort of protection against them. Besides manipulation [2,12] (also referred to as strategic voting) and electoral control [3,24], much work has been done to study bribery attacks. For a comprehensive overview of the formal models and the related

This paper extends the preliminary conference versions that appear in the proceedings of the *17th International Conference on Autonomous Agents and Multiagent Systems* (AAMAS'18) [29] and in the nonarchival website proceedings of the *International Symposium on Artificial Intelligence and Mathematics* (ISAIM'18) by presenting all proofs (some of which were omitted in the conference versions due to space limitations) in full detail, by adding new results on iterated veto and veto with runoff in Theorems 11 and 12, and by adding more illustrating examples and discussion (in particular, the discussion in Section 7 with new Theorems 13 and 14). This work was supported in part by DFG grants RO 1202/15-1, RO 1202/14-2, and BA 6270/1-1.

Heinrich-Heine-Universität Düsseldorf
Universitätsstr. 1
40225 Düsseldorf, Germany
E-mail: {cynthia.maushagen, marc.neveling, rothe, ann-kathrin.selker}@hhu.de

complexity results, we refer to the book chapters by Conitzer and Walsh [13] for manipulation, by Faliszewski and Rothe [21] for control and bribery, and by Baumeister and Rothe [5] for all three topics.

Bribery in voting was introduced by Faliszewski et al. [17] (see also [18]). In their model, a briber intends to change the outcome of an election to his or her own advantage by bribing certain voters without exceeding a given budget. Bribery shares some features with manipulation, as the briber (just like a strategic voter) has to find the right preference orders that the bribed voters are then requested to change their votes to. Bribery also shares some features with electoral control, as the briber (just like an election chair) has to pick the right voters to bribe so as to make the cost of bribing them as inexpensive as possible and to stay within the allowed budget.

We will focus on shift bribery, a special case of swap bribery, which was introduced by Faliszewski et al. [18] in the context of so-called irrational voters for Copeland elections and was then studied in detail by Elkind et al. [16] for the constructive variant (where the briber's goal is to make a favorite candidate win the election) and was later studied by Kaczmarczyk and Faliszewski [25] in the destructive variant (where the briber's goal is to make sure that a despised candidate does not win the election). In swap bribery, the briber has to pay for each swap of any two candidates in the votes. Shift bribery additionally requires that swaps always involve the designated candidate that the briber wants to see win (in the constructive case) or not win (in the destructive case).

Swap bribery generalizes the possible winner problem [27,40], which itself is a generalization of unweighted coalitional manipulation. Therefore, each of the many hardness results known for the possible winner problem is directly inherited by the swap bribery problem. This was the motivation for Elkind et al. [16] to look at restricted variants of swap bribery such as shift bribery.

Even though shift bribery possesses a number of hardness results [16], it has also been shown to allow exact and approximate polynomial-time algorithms in a number of cases [16,15,37]. For example, Elkind et al. [16] provided a 2-approximation algorithm for shift bribery when using Borda voting.[1] This result was extended by Elkind and Faliszewski [15] to all positional scoring rules; they also obtained somewhat weaker approximations for Copeland and maximin voting. Very recently Faliszewski et al. [19] further extended this result to a polynomial-time approximation scheme. For Bucklin and fallback voting, the shift bribery problem is even exactly solvable in polynomial time [37].[2] In addition, Bredereck et al. [9] were the first to analyze shift bribery in terms of *parameterized* complexity, and only recently a long-standing open problem regarding the parameterized complexity of bribery (including shift bribery) with the number of candidates as the parameter (see the survey by Bredereck et al. [8] for a deeper discussion on this problem) was solved by Knop et al. [26] for a multitude of voting rules. Furthermore, Bredereck et al. [11] introduced combinatorial shift bribery in which a single shift bribery action affects multiple voters and Bredereck et al. [10] studied shift bribery in the context of multiwinner elections for various committee selection rules.

While the complexity of shift bribery has been comprehensively investigated for many standard voting rules, it has not been considered yet for *iterative* voting systems. To close this glaring gap, we study shift bribery for eight iterative voting systems that are based on any one of the Borda, plurality, and veto rules (see Footnote 1 for their definitions) and that each proceed in rounds, eliminating after each except the last round the candidates performing worst in a certain sense:

– The *system of Baldwin* [1] eliminates the candidates with the lowest Borda score and

[1] In *Borda* with m candidates, each vote is a linear order of the candidates, the ith candidate in a vote scores $m-i$ points, and whoever has the most points wins. Borda is a very prominent positional scoring rule and can be described by the scoring vector $(m-1,m-2,\ldots,0)$. Other prominent positional scoring rules are *plurality*, where only the top candidates in the votes score a point and no one else (i.e., plurality has the scoring vector $(1,0,\ldots,0)$), and *veto* (a.k.a. *antiplurality*), where all except the bottom candidates in the votes score a point (i.e., veto has the scoring vector $(1,\ldots,1,0)$); again, whoever has the most points wins in these rules.

[2] Faliszewski et al. [20] have complemented these results on Bucklin and fallback voting. In particular, they studied a number of bribery problems for these rules, including a variant called "extension bribery," which was previously introduced by Baumeister et al. [4] in the context of campaign management when the voters' ballots are truncated.

Table 1 Summary of complexity results for shift bribery problems

	Hare	Coombs	Baldwin	Nanson
Constructive	NP-c (Thm. 1)	NP-c (Thm. 3)	NP-c (Thm. 5)	NP-c (Thm. 7)
Destructive	NP-c (Thm. 2)	NP-c (Thm. 4)	NP-c (Thm. 6)	NP-c (Thm. 8)
	Iterated Plurality	Plurality with Runoff	Iterated Veto	Veto with Runoff
Constructive	NP-c (Thm. 9)	NP-c (Thm. 9)	NP-c (Thm. 11)	NP-c (Thm. 11)
Destructive	NP-c (Thm. 10)	NP-c (Thm. 10)	NP-c (Thm. 12)	NP-c (Thm. 12)

- the *system of Nanson* [30] eliminates the candidates whose scores are lower than the average Borda score, while
- the *system of Hare* (see, e.g., the book by Taylor [38]) eliminates the candidates with the lowest plurality score,
- the system called *iterated plurality* (again see, e.g., the book by Taylor [38]) eliminates the candidates that do not have the highest plurality score,
- the system called *iterated veto* is defined analogously to iterated plurality, except based on the veto rather than the plurality score, and
- the *system of Coombs* (defined, e.g., in the paper by Levin and Nalebuff [28]) eliminates the candidates with the lowest veto score.

The last two systems that we consider differ from the above iterative voting systems because they always use exactly two rounds:

- *Plurality with runoff* (as defined, e.g., in the book by Taylor [38]) eliminates the candidates that do not have the highest plurality score, except in the case where there is a unique plurality winner—it then eliminates all candidates that do not have the highest or second-highest plurality score; all remaining candidates with the highest plurality score then win.
- *Veto with runoff* is defined analogously, except that veto scores instead of plurality scores and veto winners instead of plurality winners are considered.

These voting systems have been thoroughly studied and are also used in the real world. Among the systems we consider, Hare voting and variants thereof (some of which are called single transferable vote, instant-runoff voting, or alternative vote) are most widely used, for example in Australia, India, Ireland, New Zealand, Pakistan, the UK, and the USA.

Table 1 gives an overview of our complexity results for constructive and destructive shift bribery in our eight voting systems,[3] where the shorthand NP-c stands for "NP-complete." Our results complement results by Davies et al. [14] who have shown unweighted coalitional manipulation to be NP-complete for Baldwin and Nanson voting (even with just a single manipulator)—and also for the underlying Borda system (with two manipulators; for the latter result, see also the paper by Betzler et al. [6]). Davies et al. [14] also list various appealing features of the systems by Baldwin and Nanson, including that they have been applied in practice (namely, in the State of Michigan in the 1920s, in the University of Melbourne from 1926 through 1982, and in the University of Adelaide since 1968) and that (unlike Borda itself) they both are Condorcet-consistent.[4]

This paper is organized as follows. In Section 2, we will provide the needed definitions regarding elections and voting systems (in particular, iterative voting systems), define the shift bribery problem, and give some background on computational complexity. We will then study the complexity of shift bribery for Hare and Coombs elections in

[3] As shown by Xia [39], destructive bribery is closely related to the *margin of victory*, a critical robustness measure for voting systems. Reisch et al. [33] add to this connection by showing that the former problem can be easy while the latter is hard.

[4] A *Condorcet winner* is a candidate who defeats every other candidate in a pairwise comparison. Such a candidate does not always exist. A voting rule is *Condorcet-consistent* if it chooses only the Condorcet winner whenever there exists one.

Section 3, for Baldwin and Nanson elections in Section 4, for iterated plurality and plurality with runoff in Section 5, and for iterated veto and veto with runoff in Section 6. Further, in Section 7 we will discuss how the nonmonotonicity property of our iterative voting systems can be exploited in our reductions showing NP-hardness, exemplified for Hare voting and plurality with runoff. Finally, we will conclude in Section 8 by presenting some open problems related to our work.

2 Preliminaries

Below, we provide the needed notions and notation.

Elections and voting systems. An *election* is specified as a pair (C,V) with C being a set of candidates and V a profile of the voters' preferences over C, typically given by a list of linear orders of the candidates. A *voting system* is a function that maps each election (C,V) to a subset of C, the *winner(s) of the election*. An important class of voting systems is the family of positional scoring rules whose most prominent members are plurality, veto, and Borda count, see, e.g., the book chapters by Zwicker [41] and Baumeister and Rothe [5] and also Rothe's survey [36] on using Borda in collective decision making.

Recall from Footnote 1 in Section 1 that, in *plurality*, each voter gives her top-ranked candidate one point; in *veto* (a.k.a. *antiplurality*), each voter gives all except the bottom-ranked candidate one point; in *Borda* with m candidates, each candidate in position i of the voters' rankings scores $m-i$ points; and the winners in each case are those candidates scoring the most points.

Iterative voting systems. The iterative voting systems we will study are based on plurality, veto, and Borda but, unlike those, their election winner(s) are determined in consecutive rounds. For all iterative voting systems considered here except for plurality with runoff and veto with runoff (which will be defined shortly afterwards), if in some round all remaining candidates have the same score (for instance, there may be only one candidate left), then all those candidates are proclaimed winners of the election. In each preceding round, however, all candidates with the lowest score are eliminated.[5]

Recall from Section 1 that the eight scoring methods we will use work as follows: The iterative voting systems due to *Hare*, *Coombs*, and *Baldwin* use, respectively, plurality, veto, and Borda scores in order to decide which candidates are the weakest and thus to be removed. The *Nanson* system eliminates in every (except the last) round all candidates that have less than the average Borda score. *Iterated plurality* eliminates all candidates that do not have the highest plurality score, and *iterated veto* eliminates all candidates that do not have the highest veto score.

Unlike the above multiple-round iterative voting systems, *plurality with runoff* (respectively, *veto with runoff*) always proceeds in two rounds: In the first round, it eliminates all candidates that do not have the highest plurality score (respectively, veto score), unless there is a unique plurality winner (respectively, veto winner) in which case all candidates are eliminated except those with the highest or second-highest plurality score (respectively, veto score); in the second round, all candidates with the highest plurality score (respectively, veto score) win.

Shift bribery. For any given voting system $\mathscr{E}$, we now define the problem $\mathscr{E}$-SHIFT-BRIBERY, which is a special case of $\mathscr{E}$-SWAP-BRIBERY, introduced by Faliszewski et al. [18] in the context of so-called irrational voters for Copeland and then comprehensively studied by Elkind et al. [16]. Informally stated, given a profile of votes, a swap-bribery price function exacts a price for each swap of any two candidates in the votes, and in shift bribery only swaps involving the designated candidate are allowed.

[5] In the original sources defining these iterative voting systems as stated in the Introduction, certain tie-breaking schemes are used whenever more than one candidate has the lowest score in some round. For the sake of convenience and uniformity, however, we prefer eliminating them all and will therefore disregard tie-breaking issues in such a case.

$\mathscr{E}$-CONSTRUCTIVE-SHIFT-BRIBERY	
Given:	An election (C,V) with n votes, a designated candidate $p \in C$, a budget B, and a list of price functions $\rho = (\rho_1, \ldots, \rho_n)$.
Question:	Is it possible to make p the unique $\mathscr{E}$ winner of the election by shifting p in the votes such that the total price does not exceed B?

In the corresponding problem $\mathscr{E}$-DESTRUCTIVE-SHIFT-BRIBERY, given the same input, we ask whether it is possible to prevent p from being a unique winner.

These problems are here defined in the unique-winner model where a constructive (respectively, destructive) bribery action is considered successful only if the designated candidate can be made (respectively, can be prevented from being) the only winner of the election. We also consider these problems in the nonunique-winner model where for a constructive (respectively, destructive) bribery action to be considered successful it is required that the designated candidate is merely one among possibly several winners (respectively, does not win at all). Note that a yes-instance of $\mathscr{E}$-CONSTRUCTIVE-SHIFT-BRIBERY in the unique-winner model is also a yes-instance of the same problem in the nonunique-winner model, whereas a yes-instance of $\mathscr{E}$-DESTRUCTIVE-SHIFT-BRIBERY in the nonunique-winner model is also a yes-instance of the same problem in the unique-winner model; analogous statements apply to the no-instances of these problems by swapping the unique-winner model with the nonunique-winner model. We will make use of these facts in our proofs that all work in both winner models.

Membership in NP is obvious for all considered problems, so it will be enough to show only NP-hardness so as to prove in fact NP-completeness.

Regarding the list of price functions $\rho = (\rho_1, \ldots, \rho_n)$ with $\rho_i : \mathbb{N} \to \mathbb{N}$, in the constructive case $\rho_i(k)$ indicates the price the briber has to pay in order to move p in vote i by k positions to the top (respectively, to the bottom in the destructive case). For all i, we require that ρ_i is nondecreasing ($\rho_i(\ell) \leq \rho_i(\ell+1)$), $\rho_i(0) = 0$, and if p is at position r in vote i then $\rho_i(\ell) = \rho_i(\ell-1)$ whenever $\ell \geq r$ in the constructive case (respectively, whenever $\ell \geq |C| - r + 1$ in the destructive case). The latter condition ensures that p can be shifted upward no farther than to the top (respectively, the bottom).[6] When the voter i in ρ_i is clear from the context, we omit the subscript and simply write ρ.

Our proofs use the following notation: A vote of the form $a\ b\ c$ indicates that the voter ranks candidate a on top position, then candidate b, and last candidate c. If a set $S \subseteq C$ of candidates appears in a vote as $\overrightarrow{S}$, its candidates are placed in this position in lexicographical order. By $\overleftarrow{S}$ we mean the reverse of the lexicographical order of the candidates in S. If S occurs in a vote without an arrow on top, the order in which the candidates from S are placed here does not matter for our argument. We use $\cdots$ in a vote to indicate that the remaining candidates may occur in any order.

Computational complexity. We assume familiarity with the standard concepts of complexity theory, including the classes P and NP, polynomial-time many-one reducibility, and NP-hardness and -completeness. We will use the following NP-complete problem:

EXACT-COVER-BY-3-SETS (X3C)	
Given:	A set $X = \{x_1, \ldots, x_{3m}\}$ and a family of sets $\mathscr{S} = \{S_1, \ldots, S_n\}$ such that $S_i \subseteq X$ and $\lvert S_i \rvert = 3$ for all $S_i \in \mathscr{S}$.
Question:	Does there exist an exact cover of X, i.e., a subset $\mathscr{S}' \subseteq \mathscr{S}$ such that $\lvert \mathscr{S}' \rvert = m$ and $\bigcup_{S_i \in \mathscr{S}'} S_i = X$?

In instances of X3C, we assume that each $x_j \in X$ is contained in exactly three sets $S_i \in \mathscr{S}$; thus $|X| = |\mathscr{S}|$. Gonzalez [23] shows that X3C under this restriction remains NP-hard. Note that if not stated otherwise, we will use $(X, \mathscr{S})$ to denote an X3C instance, where $X = \{x_1, \ldots, x_{3m}\}$, $\mathscr{S} = \{S_1, \ldots, S_{3m}\}$, and $S_i = \{x_{i,1}, x_{i,2}, x_{i,3}\}$. Also note that we assume $x_{i,1}$ to be the $x_j \in S_i$ with the smallest subscript and $x_{i,3}$ to be the $x_j \in S_i$ with the largest subscript.

[6] If p is in the first (respectively, the last) position of a vote, this voter cannot be bribed and we tacitly assume a price function of $\rho(t) = 0$ for each $t \geq 0$. We will disregard these voters when setting price functions for the other voters in our proofs.

ONE-IN-THREE-POSITIVE-3SAT	
Given:	A set X of boolean variables, a set S of clauses over X, each containing exactly three unnegated literals.
Question:	Does there exist a truth assignment to the variables in X such that exactly one literal is set to true for each clause in S?

In instances of ONE-IN-THREE-POSITIVE-3SAT, we assume that each $x_j \in X$ is contained in exactly three clauses. Porschen et al. [32] show that this restricted problems remains NP-complete.

For more background on computational complexity, the reader is referred to, for instance, the textbooks by Garey and Johnson [22], Papadimitriou [31], and Rothe [34].

3 Hare and Coombs

We start by showing NP-hardness of shift bribery for Hare elections.

Theorem 1 *In both the unique-winner and the nonunique-winner model, Hare-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. NP-hardness follows by a reduction from X3C. Given an X3C instance $(X, \mathscr{S})$, construct an instance $((C,V), p, B, \rho)$ of Hare-CONSTRUCTIVE-SHIFT-BRIBERY with candidate set $C = X \cup \mathscr{S} \cup \{p\}$, designated candidate p, and the following list V of votes, with # denoting their number:

#	vote	for
1	$S_i\ x_{i,1}\ \overrightarrow{X \setminus \{x_{i,1}\}} \cdots$	$1 \le i \le 3m$
1	$S_i\ x_{i,2}\ \overrightarrow{X \setminus \{x_{i,2}\}} \cdots$	$1 \le i \le 3m$
1	$S_i\ x_{i,3}\ \overrightarrow{X \setminus \{x_{i,3}\}} \cdots$	$1 \le i \le 3m$
4	$x_i\ \overrightarrow{X \setminus \{x_i\}} \cdots$	$1 \le i \le 3m$
1	$S_i\ p\ \cdots$	$1 \le i \le 3m$
3	$p\ \cdots$	

For votes of the form $S_i\ p\ \cdots$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \ge 2$; and for every other vote, we use the price function ρ with $\rho(t) = m+1$ for all $t \ge 1$. Finally, set the budget $B = m$. Without loss of generality, we assume that $m > 1$.

Note that p scores three points while the rest of the candidates score four points each, so p is eliminated in the first round and does not win the election without bribing voters.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Hare-CONSTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose that $(X, \mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. We now show that it is possible for p to become a unique Hare winner of an election obtained by shifting p in the votes without exceeding the budget B. For every $S_i \in \mathscr{S}'$, we bribe the voter with the vote of the form $S_i\ p\ \cdots$ by shifting p once, so her new vote is of the form $p\ S_i\ \cdots$; each such bribe action costs us only 1 from our budget, so the budget will not be exceeded. In the first round, p now has $m+3$ points, every candidate from $\mathscr{S}'$ has 3 points, and every other candidate has 4 points. Therefore, all candidates in $\mathscr{S}'$ are eliminated. In the second round, all candidates in X now gain one point from the elimination of $\mathscr{S}'$, since it is an exact cover. Therefore, p and all candidates in X proceed to the next round and the remaining candidates $\mathscr{S} \setminus \mathscr{S}'$ are eliminated. In the next round with only p and

the candidates from X remaining, p has $3m+3$ points, while every candidate in X scores 7 points (recall that every $x_i \in X$ is contained in exactly three members of $\mathscr{S}$). Since all candidates from X have been eliminated now, p is the only remaining candidate and thus the unique Hare winner.

($\Leftarrow$) Suppose that $(X,\mathscr{S})$ is a no-instance of X3C. Then no subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \leq m$ covers X. We now show that we cannot make p become a Hare winner of an election obtained by bribing voters without exceeding budget B. Note that we can only bribe at most m voters with votes of the form $S_i\ p\ \cdots$ without exceeding the budget. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that $S_i \in \mathscr{S}'$ exactly if the voter with the vote $S_i\ p\ \cdots$ has been bribed. Clearly, $|\mathscr{S}'| \leq m$ and in all those votes p has been shifted once to the left, so p is now ranked first in these votes. Therefore, p now has $3+|\mathscr{S}'|$ points and every $S_i \in \mathscr{S}'$ scores 3 points. Since every other candidate scores as many points as before the bribery (namely, 4 points), the candidates in $\mathscr{S}'$ are eliminated in the first round. Let $X' = \{x_i \in X \mid x_i \notin \bigcup_{S_j \in \mathscr{S}'} S_j\}$ be the subset of candidates $x_i \in X$ that are not covered by $\mathscr{S}'$. We have $X' \neq \emptyset$ (otherwise, $\mathscr{S}'$ would have been an exact cover of X). In the second round, unlike the candidates from $X \setminus X'$, the candidates in X' will not gain additional points from eliminating the candidates in $\mathscr{S}'$. Thus, in the current situation, the candidates from X' and $\mathscr{S} \setminus \mathscr{S}'$ are trailing behind with 4 points each and are eliminated in this round.[7] Therefore, in the next round, only p and the candidates from $X \setminus X'$ are remaining in the election. Let $x_\ell \in X \setminus X'$ be the candidate from $X \setminus X'$ with the smallest subscript. Since all candidates from $\mathscr{S}$ are eliminated, p has $3m+3$ points and every candidate from $X \setminus X'$ except x_ℓ has 7 points. On the other hand, x_ℓ gains additional points from eliminating the candidates from X'; therefore, x_ℓ survives this round by scoring more than 7 points. In the final round with only p and x_ℓ remaining, p is eliminated, since $3m \cdot 7 > 3m+3$. □

Example 1 Let $(X,\mathscr{S})$ be a yes-instance of X3C defined by

$$X = \{x_1,\ldots,x_6\} \text{ and}$$
$$\mathscr{S} = \{\{1,2,3\},\{4,5,6\},\{2,3,6\},\{2,4,5\},\{1,3,4\},\{1,5,6\}\}.$$

Construct $((C,V),p,B,\rho)$ from $(X,\mathscr{S})$ as in the proof of Theorem 1; in particular, the budget is $B=2$. If we bribe the voters with $S_1\ p\ \cdots$ and $S_2\ p\ \cdots$ so as to shift p to the top of their votes, p will be the unique winner of the election that proceeds as follows (where the numbers in the columns below candidates give their scores):

Round	p	$x \in X$	S_1, S_2	S_3, S_4, S_5, S_6
1	5	4	3	4
2	5	5	out	4
3	9	7	out	out

Now consider a no-instance $(X,\mathscr{S})$ of X3C with

$$X = \{x_1,\ldots,x_6\} \text{ and}$$
$$\mathscr{S} = \{\{1,2,4\},\{4,5,6\},\{2,3,6\},\{2,3,5\},\{1,3,4\},\{1,5,6\}\}.$$

If we bribe no voter, p gets eliminated in the first round and so does not win. If we bribe one voter, say the one with vote $S_1\ p\ \cdots$, then p gets eliminated in the second round:

Round	p	x_1	$x_2,\ x_4$	$x_3,\ x_5,\ x_6$	S_1	$S_i \in \mathscr{S} \setminus \{S_1\}$
1	4	4	4	4	3	4
2	4	5	5	4	out	4
3	out	≥ 28	≥ 7	out	out	out

[7] Note that in the case that $|\mathscr{S}'| = 1$, i.e., only one voter was bribed, p also gets eliminated in this round and is consequently not a Hare winner, which is what we want to show. Therefore, we will now assume that at least two voters were bribed.

Since $(X, \mathscr{S})$ is a no-instance of X3C, no matter which two subsets $S_i, S_j \in \mathscr{S}$ we choose, at least one x_k is in both subsets, so p loses the direct comparison in the last round. For example, if we bribe the voters with $S_1\ p\ \cdots$ and $S_2\ p\ \cdots$, the election proceeds as follows:

Round	p	x_1	x_3	x_4	x_2, x_5, x_6	$S_1,\ S_2$	S_3, S_4, S_5, S_6
1	5	4	4	4	4	3	4
2	5	5	4	6	5	out	4
3	9	14	out	7	7	out	out
4	9	42	out	out	out	out	out

This completes Example 1.

Next, we show that shift bribery is NP-hard for Hare also in the destructive case.

Theorem 2 *In both the unique-winner and the nonunique-winner model, Hare-*DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. Again, we use a reduction from X3C. Construct from a given X3C instance $(X, \mathscr{S})$ a Hare-DESTRUCTIVE-SHIFT-BRIBERY instance $((C,V), p, B, \rho)$ as follows. Let $D = \{d_1, \ldots, d_{3m}\}$ be a set of $3m$ dummy candidates. The candidate set is $C = X \cup \mathscr{S} \cup D \cup \{p, w\}$ with designated candidate p. The list V of votes is constructed as follows:

#	vote	for
2	$S_i\ x_{i,1}\ \overrightarrow{X \setminus \{x_{i,1}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
2	$S_i\ x_{i,2}\ \overrightarrow{X \setminus \{x_{i,2}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
2	$S_i\ x_{i,3}\ \overrightarrow{X \setminus \{x_{i,3}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
7	$x_i\ \overrightarrow{X \setminus \{x_i\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
1	$p\ S_i\ \cdots$	$1 \le i \le 3m$
12	$w\ p\ \cdots$	
$18m$	$p\ \cdots$	
6	$d_i\ S_i\ p\ \cdots$	$1 \le i \le 3m$

For votes of the form $p\ S_i\ \cdots$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \ge 2$; and for every other vote, we use the price function ρ with $\rho(t) = m+1$ for all $t \ge 1$. Finally, set the budget $B = m$.

Without bribing, the election (C,V) proceeds as follows:

Round	p	w	$x_i \in X$	$S_i \in \mathscr{S}$	$d_i \in D$
1	$21m$	12	7	6	6
2	$39m$	12	13	out	out
3	$39m+12$	out	13	out	out

It follows that p has won the election after three rounds.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Hare-DESTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose that $(X, \mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. We now show that it is possible to eliminate p from an election obtained by shifting p in the votes without exceeding the

budget B. For every $S_i \in \mathscr{S}'$, we bribe the voter with the vote of the form $p\ S_i\ \cdots$ by shifting p once, so her new vote is of the form $S_i\ p\ \ldots$; each such bribe action costs us only 1 from our budget, so the budget will not be exceeded. Now the election proceeds as follows:

Round	p	w	$x_i \in X$	$S_i \in \mathscr{S}'$	$S_i \in \mathscr{S} \setminus \mathscr{S}'$	$d_i \in D$
1	$20m$	12	7	7	6	6
2	$32m$	12	11	13	out	out
3	$32m$	$33m+12$	out	13	out	out
4	$39m$	$39m+12$	out	out	out	out

We see that p is eliminated in the fourth round and w wins.

($\Leftarrow$) Suppose that $(X,\mathscr{S})$ is a no-instance of X3C. Then no subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \leq m$ covers X. We now show that p will not be eliminated in any election obtained by bribing voters without exceeding budget B but will in fact become the only winner. Note that we can only bribe at most m voters with votes of the form $p\ S_i\ \cdots$ without exceeding the budget. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that for every $S_i \in \mathscr{S}'$ we have bribed the voter whose vote is $p\ S_i\ \cdots$. We can assume that $|\mathscr{S}'| > 0$. Every candidate in $\mathscr{S}'$ will gain an additional point and therefore survives the first round. All candidates from D and $\mathscr{S} \setminus \mathscr{S}'$ will be eliminated, since p only loses at most m points.

In the second round, the remaining candidates from $\mathscr{S}$ will additionally gain six points from the elimination of candidates in D and will score 13 points in this round (and in all subsequent rounds with p still standing). If a candidate $S_i \in \mathscr{S}$ was eliminated in the previous round, every $x_i \in S_i$ gains two additional points in this round. Partition X into sets X_0, X_1, X_2, and X_3 so that $x_i \in X_k \Leftrightarrow |\{S_j \in \mathscr{S}' \,|\, x_i \in S_j\}| = k$ for $k \in \{0,1,2,3\}$. Note that X_0, X_1, X_2, and X_3 are disjoint and $|X_0| > 0$, but one or two of X_1, X_2, and X_3 may be empty. Then $x_i \in X_j$ scores $7+(6-2j) \in \{7,9,11,13\}$ points depending on how many times x_i is covered by $\mathscr{S}'$. Therefore, every $x_i \in X_0$ scores more points than w who has 12 points. Thus there are candidates from X that survive this round and other candidates from X (more precisely, candidates from X_1, X_2, or X_3) who are eliminated.

In the third round, the candidate $x_\ell \in X$ with the smallest subscript who is still standing gains at least seven points from the eliminated candidates, so that x_ℓ scores at least 16 points.[8] All other candidates still score the same number of points as in the last round. Therefore, p scores at least $20m$ points, w scores still 12 points, every $S_i \in \mathscr{S}'$ scores 13 points, and every still standing candidate from X except x_ℓ scores at most 13 points. Since w can only gain additional points when all candidates from X are eliminated and only x_ℓ gains points from the elimination of candidates from $X \setminus \{x_\ell\}$ in the subsequent rounds, all candidates $X \setminus (\{x_\ell\} \cup X_0)$ and w are eliminated. Then all still standing candidates from $X_0 \setminus \{x_\ell\}$ and candidates from $\mathscr{S}'$ who each score 13 points are eliminated, which leaves p and x_ℓ in the last round. In this round, p scores $39m+12$ points and x_ℓ scores $39m$ points, so p solely wins the election, no matter how we bribe voters within the budget, i.e., we have a no-instance of Hare-DESTRUCTIVE-SHIFT-BRIBERY in both winner models. □

Next, we turn to shift bribery for Coombs elections. While the idea of the reduction is similar, and perhaps even simpler than in the previous two proofs, the proof of correctness is way more involved.

Theorem 3 *In both the unique-winner and the nonunique-winner model, Coombs-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we now describe a reduction from X3C to Coombs-CONSTRUCTIVE-SHIFT-BRIBERY. Given an X3C instance $(X,\mathscr{S})$, construct an election (C,V) with the set $C = \{p,w,d_1,d_2,d_3\} \cup X \cup Y$ of candidates, where p is the designated candidate and $Y = \{y_i \,|\, x_i \in X\}$. Construct the following list V of votes:

[8] Since this candidate x_ℓ is still in the election, x_ℓ cannot have been in X_3 and thus must have had at least nine points.

#	vote	for
1	$\cdots\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p$	$1 \le i \le 3m$
$2m$	$\cdots\ p\ \overrightarrow{Y \setminus \{y_i\}}\ y_i\ x_i$	$1 \le i \le 3m$
$2m$	$\cdots\ p\ \overrightarrow{Y}\ w\ d_1\ d_2\ d_3$	
1	$\cdots\ p\ \overrightarrow{Y}\ w\ X\ d_1\ d_2\ d_3$	
m	$\cdots\ p\ \overrightarrow{Y}\ w$	

For votes of the form $\cdots\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p$, we use the price function $\rho(1) = \rho(2) = \rho(3) = 1$, and $\rho(t) = m+1$ for all $t \ge 4$; and for all the remaining votes, we use the price function $\rho(t) = m+1$ for all $t \ge 1$. Furthermore, our budget is $B = m$.

The candidates have the following veto counts: p has $3m$ vetoes, each $x_i \in X$ has $2m$ vetoes, w has m vetoes, d_3 has $2m+1$ vetoes, and the remaining candidates each have 0 vetoes. Therefore, p will be eliminated in the first round and thus does not win the election.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Coombs-CONSTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Assume that $(X, \mathscr{S})$ is in X3C. This means that there exists a subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| = m$ and $\bigcup_{S_i \in \mathscr{S}'} S_i = X$. So we have a partition of X into three sets, $X = X_1 \cup X_2 \cup X_3$, such that:

$$X_1 = \{x_i \in S_i \mid x_i \text{ has the lowest subscript in } S_i \in \mathscr{S}'\},$$
$$X_3 = \{x_i \in S_i \mid x_i \text{ has the highest subscript in } S_i \in \mathscr{S}'\}, \text{ and}$$
$$X_2 = X \setminus (X_1 \cup X_3).$$

Let $Y = Y_1 \cup Y_2 \cup Y_3$ be the corresponding partition of Y.

We bribe the voters with votes of the form $\cdots\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p$ for $S_i \in \mathscr{S}'$ so that they change their votes to $\cdots\ p\ x_{i,1}\ x_{i,2}\ x_{i,3}$. Since $\mathscr{S}'$ is an exact cover of X, it follows that p now has a total of $2m$ vetoes, whereas each $x \in X_3$ receives an additional veto for a total of $2m+1$. The number of vetoes for the remaining candidates remain unchanged. If a candidate has the highest number of vetoes then she has the fewest number of points and cannot proceed to the next round (unless all candidates have the same score). Here, the candidates in X_3 and d_3 have the fewest number of points (and fewer than the other candidates) and therefore are eliminated in the first round.

Without the candidates in X_3, each candidate in X_2 gets an additional veto and the candidates in Y_3 each take all but one of the vetoes of the eliminated candidates in X_3. Furthermore, d_2 receives the vetoes of d_3. As a consequence, in the second round the candidates in X_2 and d_2 have the fewest number of points (and fewer than the remaining candidates) and are eliminated.

Similarly to the first round, vetoes from candidates in X_2 and d_2 are passed on to candidates in X_1 and Y_1 and to d_1. Thus the candidates have the following veto counts in the third round: p and each $y \in Y_2 \cup Y_3$ receive $2m$ vetoes, w receives m vetoes, each $y \in Y_1$ receives zero vetoes, and d_1 and each $x_i \in X_1$ receive $2m+1$ vetoes. Consequently, all the candidates $x_i \in X_1$ and d_1 are eliminated in the third round, so in the next round there are no candidates from X and no d_i, $1 \le i \le 3$.

It follows that w receives $2m+1$ additional vetoes in the fourth round, so w has the most vetoes in this round and is eliminated. We need $3m$ further rounds until p ends up as the last remaining candidate and sole winner of the election. In each of these rounds, the candidate in Y that is still alive and has the highest subscript has at least $2m+2m+1+m = 5m+1$ vetoes, while p always has only $3m$ vetoes.

($\Leftarrow$) Suppose that $(X, \mathscr{S})$ is a no-instance of X3C. We will show that $((C,V), p, B, \rho)$ then is a no-instance of Coombs-CONSTRUCTIVE-SHIFT-BRIBERY in the nonunique-winner (and thus also in the unique-winner) model. Observe that if we were going to make p a winner of the election, we would have to bribe at least m voters with a

vote of the form $\cdots\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p$; otherwise, p would have at least $2m+1$ vetoes and would be eliminated right away in the first round. Due to our budget, on the other hand, we can bribe no more than m (and thus would have to bribe exactly m) such voters and cannot bribe any further voters. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that $S_i \in \mathscr{S}'$ exactly if the voter with the vote of the form $\cdots\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p$ has been bribed. Note that $|\mathscr{S}'| = m$ and $\mathscr{S}'$ does not cover X because we have a no-instance of X3C. Now p has only $2m$ vetoes and will not be eliminated in the first round.

Let X_1 be the set of candidates $x_i \in S_i$ for $S_i \in \mathscr{S}'$ with the smallest subscript in S_i, let X_2 be the set of candidates $x_i \in S_i$ for $S_i \in \mathscr{S}'$ with the second-smallest subscript in S_i, and let X_3 be the set of candidates $x_i \in S_i$ for $S_i \in \mathscr{S}'$ with the highest subscript in S_i. Note that $X_1 \cup X_2 \cup X_3 \neq X$, since $\mathscr{S}'$ does not cover X.

For w to have more vetoes than p, the candidates d_1, d_2, and d_3 need to be eliminated. For that to happen, there must be three rounds in which no other candidate has more than $2m+1$ vetoes. In the round where d_i, $1 \leq i \leq 3$, is eliminated, all still standing candidates in X_i are eliminated as well. Assume there were three rounds in which $2m+1$ was the maximal number of vetoes for a candidate. Then d_1, d_2, d_3, and all candidates in $X_1 \cup X_2 \cup X_3$ are eliminated. Note that those candidates that are not covered by $\mathscr{S}'$ always have only $2m$ vetoes and are still participating in the election. Therefore, in the next round, p and w have $3m$ vetoes each, the remaining candidates from X have at most $2m+1$ vetoes, and the candidates from Y have at most $2m$ vetoes. So even if p survives the first rounds with the candidates d_1, d_2, and d_3 still present, p will then surely be eliminated in the following round. If there is at least one voter who shifts p only one or two positions upward, then p has to drop out with d_1 or even before d_1 drops out, because at the latest after two rounds (with $2m+1$ being the maximal number of vetoes for a candidate) p receives another veto and thus has at least the same number of vetoes as d_1. □

Example 2 Let $(X,\mathscr{S})$ be a yes-instance of X3C defined by

$$X = \{x_1,\ldots,x_6\} \text{ and}$$
$$\mathscr{S} = \{\{1,2,3\},\{4,5,6\},\{2,3,6\},\{2,4,5\},\{1,3,4\},\{1,5,6\}\}.$$

Construct $((C,V),p,B,\rho)$ from $(X,\mathscr{S})$ as in the proof of Theorem 3; in particular, the budget is $B=2$. If we bribe the voters that correspond to the sets in the exact cover, S_1 and S_2, to change their votes from $\cdots\ x_1\ x_2\ x_3\ p$ and $\cdots\ x_4\ x_5\ x_6\ p$ to $\cdots p\ x_1\ x_2\ x_3$ and $\cdots p\ x_4\ x_5\ x_6$, then p alone wins the election that proceeds as follows, where in order to make this example easier to follow, the numbers in the table count the candidates' vetoes, not their points, i.e., the candidates with the *highest* number in a round (row) get eliminated:

Round	p	w	x_1,x_4	x_2,x_5	x_3,x_6	y_1	y_2	y_3	y_4	y_5	y_6	d_1	d_2	d_3
1	4	2	4	4	5	0	0	0	0	0	0	0	0	5
2	4	2	4	5	out	0	0	4	0	0	4	0	5	out
3	4	2	5	out	out	0	4	4	0	4	4	5	out	out
4	6	7	out	out	out	4	4	4	4	4	4	out	out	out
5	6	out	out	out	out	4	4	4	4	4	11	out	out	out
6	6	out	out	out	out	4	4	4	4	15	out	out	out	out
7	6	out	out	out	out	4	4	4	19	out	out	out	out	out
8	6	out	out	out	out	4	4	23	out	out	out	out	out	out
9	6	out	out	out	out	4	27	out	out	out	out	out	out	out
10	6	out	out	out	out	31	out	out	out	out	out	out	out	out

It follows that p is the sole winner of the election.

Now consider a no-instance $(X,\mathscr{S})$ with

$$X = \{x_1,\ldots,x_6\} \text{ and}$$
$$\mathscr{S} = \{\{1,2,4\},\{4,5,6\},\{2,3,6\},\{2,3,5\},\{1,3,4\},\{1,5,6\}\}.$$

Recall that we can bribe at most two voters. If we bribe fewer than two voters, however, p will be eliminated in the first round. Since $(X, \mathscr{S})$ is a no-instance of X3C, no matter which two subsets $S_i, S_j \in \mathscr{S}$ we choose, at least one x_k is in both S_i and S_j. For example, if we bribe the voters that correspond to the sets S_1 and S_2, changing their votes from $\cdots\ x_1\ x_2\ x_4\ p$ and $\cdots\ x_4\ x_5\ x_6\ p$ to $\cdots p\ x_1\ x_2\ x_4$ and $\cdots p\ x_4\ x_5\ x_6$, then the election proceeds as follows:

Round	p	w	x_1	x_2, x_5	x_3	x_4, x_6	y_1	y_2, y_5	y_3	y_4, y_6	d_1	d_2	d_3
1	4	2	4	4	4	5	0	0	0	0	0	0	5
2	4	2	4	5	4	out	0	0	0	4	0	5	out
3	5	2	5	out	4	out	0	4	0	4	5	out	out
4	out	2	out	out	4	out	4	4	0	4	out	out	out
...	...	...	...	...	...	...	...	...	...	...	...	...	...

Since x_4 is in both S_1 and S_2, p gets an additional veto in round 3 and is subsequently eliminated. The same will happen for similar reasons in every other case.

This completes Example 2.

We now modify the previous reduction so as to work for the destructive case in Coombs elections.

Theorem 4 *In both the unique-winner and the nonunique-winner model, Coombs-*DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we again reduce from the NP-complete problem X3C to Coombs-DESTRUCTIVE-SHIFT-BRIBERY. Given an X3C instance $(X, \mathscr{S})$ where we may assume that $m > 2$ for $|X| = 3m$, we construct a DESTRUCTIVE-SHIFT-BRIBERY instance $((C,V), p, B, \rho)$ as follows. Let $C = X \cup \mathscr{S} \cup D \cup \{p, w, y\}$ be the candidate set with designated candidate p and a set $D = \{d_{i,j} \mid 1 \leq i \leq m-1, 1 \leq j \leq 4\}$ of dummy candidates. Let $D = D_1 \cup D_2 \cup D_3 \cup D_4$ be a partition of D with $D_j = \{d_{i,j} \mid 1 \leq i \leq m-1\}$ for $1 \leq j \leq 4$. The list V of votes is then constructed as follows:

#	vote	for
1	$\cdots\ p\ S_i$	$1 \leq i \leq 3m$
$4m$	$p\ \cdots\ w\ x_{i,1}\ x_{i,2}\ x_{i,3}\ S_i$	$1 \leq i \leq 3m$
$4m+1$	$\cdots\ p\ X\ d_{i,1}\ d_{i,2}\ d_{i,3}\ d_{i,4}$	$1 \leq i \leq m-1$
1	$p\ \cdots\ y\ x_i$	$1 \leq i \leq 3m$
3	$\cdots\ p$	
2	$p\ \cdots\ w$	

Unlike in the previous proofs, it is here necessary that the candidates that are represented by "$\cdots$" are placed in lexicographical order. For votes of the form $\cdots\ p\ S_i$, we use the price function $\rho(1) = 1$, and $\rho(t) = 2m+1$ for all $t \geq 2$; and for all the remaining voters, we use the price function $\rho(t) = 2m+1$ for all $t \geq 1$. Finally, we set the budget $B = 2m$.

Analyzing the constructed election without bribing voters, the candidates have the following veto counts: p has three vetoes, w has two vetoes, each $x \in X$ has one veto, each $S_i \in \mathscr{S}$ and each $d \in D_4$ has $4m+1$ vetoes, and the remaining candidates each have zero vetoes. It follows that all candidates from $\mathscr{S}$ and D_4 are eliminated. The candidates from D_4 transfer their vetoes to candidates in D_3 who each have $4m+1$ vetoes now; p gets $3m$ additional vetoes from the eliminated candidates in $\mathscr{S}$; and the remaining $12m^2$ vetoes (from the second group of voters) are shared among candidates from X. Since they are ordered lexicographically in those votes, there must be one candidate from X (now and in the following rounds) that obtains more than $4m+1$ vetoes leading to the elimination of all candidates from X in the following rounds. In each of these following rounds, the candidate who receives some

of those $12m^2$ vetoes from a previously eliminated candidate (starting with w) will now be eliminated, eventually leaving p as the last standing candidate and sole winner.

We claim that $(X,\mathscr{S})$ is in X3C if and only if $((C,V),p,B,\rho)$ is in Coombs-DESTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Assume that $(X,\mathscr{S})$ is in X3C. This means that there exists a subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| = m$ and $\bigcup_{S_i \in \mathscr{S}'} S_i = X$. So we have a partition of X into three sets, $X = X_1 \cup X_2 \cup X_3$, such that:

$$\begin{aligned} X_1 &= \{x_i \in S_i \mid \ x_i \text{ has the lowest subscript in } S_i \in \mathscr{S}'\}, \\ X_3 &= \{x_i \in S_i \mid \ x_i \text{ has the highest subscript in } S_i \in \mathscr{S}'\}, \text{ and} \\ X_2 &= X \setminus (X_1 \cup X_3). \end{aligned}$$

We bribe the voters with a vote of the form $\cdots\ p\ S_i$ with $S_i \in \mathscr{S} \setminus \mathscr{S}'$ such that they change their vote to $\cdots\ S_i\ p$. Now the election proceeds as follows, where we again count the vetoes and not the points:

Round	p	w	y	$\mathscr{S}'$	$\mathscr{S} \setminus \mathscr{S}'$	X_1	X_2	X_3	D_1	D_2	D_3	D_4
1	$2m+3$	2	0	$4m+1$	$4m$	1	1	1	0	0	0	$4m+1$
2	$3m+3$	2	0	out	$4m$	1	1	$4m+1$	0	0	$4m+1$	out
3	$3m+3$	2	m	out	$4m$	1	$4m+1$	out	0	$4m+1$	out	out
4	$3m+3$	2	$2m$	out	$4m$	$4m+1$	out	out	$4m+1$	out	out	out
5	$4m^2+2$	$4m^2+2$	$3m$	out	$4m$	out	out	out	out	out	out	out

We see that p is eliminated in the fifth round, whereas y and some other candidates from $\mathscr{S} \setminus \mathscr{S}'$ are still in the election. Hence, p does not win.

($\Leftarrow$) Suppose that $(X,\mathscr{S})$ is a no-instance of X3C. Then no subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \leq m$ covers X. We now show that p will not be eliminated in an election obtained by bribing voters without exceeding budget B but will in fact become the only winner. Note that we can only bribe at most $2m$ voters with votes of the form $\cdots\ p\ S_i$ without exceeding the budget. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that for every $S_i \in \mathscr{S} \setminus \mathscr{S}'$ we have bribed the voter whose vote was $\cdots\ p\ S_i$ and now is $\cdots\ S_i\ p$. We can assume that $|\mathscr{S} \setminus \mathscr{S}'| > 0$.

Every candidate in $\mathscr{S} \setminus \mathscr{S}'$ will gain an additional point and therefore survives the first round. All candidates in D_4 and $\mathscr{S}'$ will be eliminated in the first round. It follows that p has $3m+3$ vetoes in the second round. At this point, p is in each voter group other than the third voter group (with votes of the form $\cdots\ p\ X\ d_{i,1}\ d_{i,2}\ d_{i,3}\ d_{i,4}$) either the most (groups 2, 4, and 6) or the least preferred (groups 1 and 5) candidate; therefore, p does not receive any further vetoes before some candidate $d \in D_1$ is eliminated.

We note that $|\mathscr{S}'| \geq m$. Since $\mathscr{S}'$ is not an exact cover of X, we have at least one $x \in X$ which is in two sets $S, S' \in \mathscr{S}'$. Let $X' = \{x \in X \mid \exists\, S,S' \in \mathscr{S}',\ S \neq S',\ x \in S \cap S'\}$. After two further rounds in which $4m+1$ is the maximum number of vetoes, the candidates $d \in D \setminus D_1$ are eliminated. If each $x \in X'$ is still in the election, it follows that each $x \in X'$ has at least $4m+2$ vetoes such that some candidates $x \in X'$ will be eliminated. It follows that in the next round w receives at least $4m+2$ vetoes such that w has the most vetoes while the candidates $d \in D_1$ still have $4m+1$ vetoes. Otherwise, if at least one candidate $x \in X'$ is eliminated, it follows that w receives at least $4m+2$ vetoes at the latest in the fourth round, while each $d \in D_1$ still has $4m+1$ vetoes. After w is eliminated, in each following round the candidate x with the highest subscript and later the candidate S with the highest subscript and y will be eliminated. It follows that only p and the candidates $d \in D_1$ are still in the election. In each following round, p has at most $4m^2 - 4m + 1$ vetoes while the still standing candidate $d \in D_1$ with the highest subscript receives at least $12m^2 + 7m + 3$ vetoes. Hence, eventually p alone wins the election. □

4 Baldwin and Nanson

We now show NP-hardness of shift bribery for Baldwin and Nanson elections. Note that our reductions are inspired by and similar to those used by Davies et al. [14] to show NP-hardness of the unweighted coalitional manipulation problem for these voting systems.

For a preference profile V over a set of candidates C, let $avg(V)$ be the average Borda score of the candidates in V (i.e., $avg(V) = (|C|-1)|V|/2$). To conveniently construct votes, for a set of candidates C and $c_1, c_2 \in C$, let

$$W_{(c_1,c_2)} = (c_1\ c_2\ \overrightarrow{C \setminus \{c_1,c_2\}},\ \overleftarrow{C \setminus \{c_1,c_2\}}\ c_1\ c_2).$$

Under Borda, from the two votes in $W_{(c_1,c_2)}$ candidate c_1 scores $|C|$ points, c_2 scores $|C|-2$ points, and all other candidates score $|C|-1$ points. Also, observe that if a candidate $c^* \in C$ is eliminated in some round and $c^* \notin \{c_1,c_2\}$ then all other candidates lose one point due to the votes in $W_{(c_1,c_2)}$; if $c^* = c_1$ then c_2 loses no points but all other candidates lose one point; and if $c^* = c_2$ then c_1 loses two points and all other candidates lose one point. Therefore, if c^* is eliminated, the point difference caused by this elimination with respect to the votes in $W_{(c_1,c_2)}$ remains the same for all candidates, with two exceptions: (a) If $c^* = c_1$ then c_2 gains a point with respect to every other candidate, and (b) if $c^* = c_2$ then c_1 loses a point with respect to every other candidate. Furthermore, let $score_{(C,V)}(x)$ denote the number of points candidate x obtains in a Borda election (C,V), and let $dist_{(C,V)}(x,y) = score_{(C,V)}(x) - score_{(C,V)}(y)$.

We start with the complexity of shift bribery in Baldwin elections for the constructive case.

Theorem 5 *In both the unique-winner and the nonunique-winner model, Baldwin-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce the NP-complete problem X3C to Baldwin-CONSTRUCTIVE-SHIFT-BRIBERY. From a given X3C instance $(X,\mathscr{S})$, we construct an election (C,V) with the set of candidates $C = \{p,w,d\} \cup X \cup \mathscr{S}$ and designated candidate p and with V consisting of two lists of votes, V_1 and V_2, where V_1 contains the following votes:

#	votes	for	#	votes	for
1	$W_{(S_j,p)}$	$1 \le j \le 3m$	2	$W_{(x_{j,3},S_j)}$	$1 \le j \le 3m$
2	$W_{(x_{j,1},S_j)}$	$1 \le j \le 3m$	2	$W_{(w,x_i)}$	$1 \le i \le 3m$
2	$W_{(x_{j,2},S_j)}$	$1 \le j \le 3m$	7	$W_{(w,p)}$	

The votes in V_1 give the following scores to the candidates in C:

$$\begin{aligned}
score_{(C,V_1)}(x_i) &= avg(V_1) + 4 \text{ for every } x_i \in X,\\
score_{(C,V_1)}(S_j) &= avg(V_1) - 5 \text{ for every } S_j \in \mathscr{S},\\
score_{(C,V_1)}(p) &= avg(V_1) - 3m - 7,\\
score_{(C,V_1)}(w) &= avg(V_1) + 6m + 7,\\
score_{(C,V_1)}(d) &= avg(V_1).
\end{aligned}$$

Furthermore, V_2 contains the following votes:

#	votes	for	#	votes
$2m+1$	$W_{(d,S_j)}$	$1 \le j \le 3m$	1	$W_{(p,d)}$
$2m+9$	$W_{(d,x_i)}$	$1 \le i \le 3m$	$2m+14$	$W_{(d,w)}$

The votes in V_2 give the following scores to the candidates in C:

$$\begin{aligned}
score_{(C,V_2)}(x_i) &= avg(V_2) - (2m+9) \text{ for every } x_i \in X,\\
score_{(C,V_2)}(S_j) &= avg(V_2) - (2m+1) \text{ for every } S_j \in \mathscr{S},\\
score_{(C,V_2)}(p) &= avg(V_2) + 1,\\
score_{(C,V_2)}(w) &= avg(V_2) - (2m+14),\\
score_{(C,V_2)}(d) &= avg(V_2) + 12m^2 + 32m + 13.
\end{aligned}$$

Let $V = V_1 \cup V_2$ and $avg(V) = avg(V_1) + avg(V_2)$. Then we have the following Borda scores for the complete preference profile V over C:

$$\begin{aligned}
score_{(C,V)}(x_i) &= avg(V) - 2m - 5 \text{ for every } x_i \in X,\\
score_{(C,V)}(S_j) &= avg(V) - 2m - 6 \text{ for every } S_j \in \mathscr{S},\\
score_{(C,V)}(p) &= avg(V) - 3m - 6,\\
score_{(C,V)}(w) &= avg(V) + 4m - 7,\\
score_{(C,V)}(d) &= avg(V) + 12m^2 + 32m + 13.
\end{aligned}$$

Regarding the price function, for every first vote of $W_{(S_j,p)}$ (i.e., a vote of the form $S_j\ p\ \overrightarrow{C \setminus \{S_j,p\}}$), let $\rho(1) = 1$ and $\rho(t) = m+1$ for every $t \geq 2$. For every other vote, let $\rho(t) = m+1$ for every $t \geq 1$. Finally, we set the budget $B = m$.

It is easy to see that p is eliminated in the first round in the election (C,V) and thus does not win.

We claim that $(X,\mathscr{S})$ is in X3C if and only if $((C,V),p,B,\rho)$ is in Baldwin-CONSTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose there is an exact cover $\mathscr{S}' \subseteq \mathscr{S}$. Then we bribe the first votes of $W_{(S_j,p)}$ for every $S_j \in \mathscr{S}'$ by shifting p to the left once. Note that we won't exceed our budget, since shifting once costs 1 in those votes and $|\mathscr{S}'| = m$. After this bribery, for every $S_j \in \mathscr{S}'$, the two votes from $W_{(S_j,p)}$ result in two votes that are symmetric to each other (i.e., $p\ S_j\ \overrightarrow{C \setminus \{S_j,p\}}$ equals the vote $\overleftarrow{C \setminus \{S_j,p\}}\ S_j\ p$ in reverse order) and can thus be disregarded from now on, as all candidates gain the same number of points from those votes and all candidates lose the same number of points if a candidate is eliminated from the election. After those m votes have been bribed, only the scores of p and every $S_j \in \mathscr{S}'$ change. With $score_{(C,V)}(p) = avg(V) - 2m - 6$ and $score_{(C,V)}(S_j) = avg(V) - 2m - 7$, all candidates in $\mathscr{S}'$ are tied for the last place. If any $S_j \in \mathscr{S}'$ is eliminated in a round, the three candidates $x_{j,1}$, $x_{j,2}$, and $x_{j,3}$ will lose two points more than the candidates from $\mathscr{S}' \setminus \{S_j\}$ that were in the last position before S_j was eliminated. Therefore, those three candidates from X will then be in the last position in the next round. This means that all candidates $\mathscr{S}'$ and every $x_i \in X$ that is covered by $\mathscr{S}'$ will be eliminated in the subsequent rounds. Since $\mathscr{S}'$ is an exact cover, now there is no candidate from X left. Thus the point difference between p and w is 1 and between p and the remaining $S_j \in (\mathscr{S} \setminus \mathscr{S}')$ is -6. Note that p can beat d only if no candidate of $C \setminus \{p,d\}$ is still participating. So in the next round, w is eliminated. From this p gains seven points on all $S_j \in (\mathscr{S} \setminus \mathscr{S}')$, so these are tied for the last place. Therefore, the remaining candidates from $\mathscr{S}$ are eliminated, which leaves p and d for the next and final round, where d is eliminated and p wins the election alone.

($\Leftarrow$) Suppose there is no exact cover. It is obvious that at most m of the first votes of $W_{(S_j,p)}$ can be bribed without exceeding the budget. Without bribing, p is in the last place and the point difference to the second-to-last candidate(s) is $dist_{(C,V)}(p,S_j) = m$, $1 \leq j \leq 3m$. By bribing, as explained above, p gains $m+1$ points on m candidates from $\mathscr{S}$, which then will be eliminated from the election. This leads to the elimination of all $x_i \in X$ that are covered

by the set $\mathscr{S}' \subseteq \mathscr{S}$ of candidates that were eliminated. Since there is no exact cover, $\mathscr{S}'$ doesn't cover X. So there are candidates $X' \subseteq X$, $|X'| \geq 1$, who were not eliminated before, as for every candidate $x_i \in X'$ all three candidates $S_j \in (\mathscr{S} \setminus \mathscr{S}')$ with $x_i \in S_j$ are still in the election. With the candidates $C_1 = \{p,w,d\} \cup (\mathscr{S} \setminus \mathscr{S}') \cup X'$ still standing, the point differences of p to the other remaining candidates are as follows:

$$
\begin{aligned}
&dist_{(C_1,V)}(p,d) = -2m-5-2m(2m+1)-|X'|(2m+9)-(2m+14) < 0,\\
&dist_{(C_1,V)}(p,w) = 1-2|X'| < 0,\\
&dist_{(C_1,V)}(p,x_i) = -1 \text{ for every } x_i \in X', \text{ and}\\
&dist_{(C_1,V)}(p,S_j) \leq 0 \text{ for every } S_j \in \mathscr{S} \setminus \mathscr{S}'.
\end{aligned}
$$

Therefore, p is in the last place and is eliminated and thus does not win. □

The proof of the following theorem, which handles the destructive variant for Baldwin, uses a similar idea as the proof of Theorem 5. That is why we refrain from presenting all proof details in full; a proof sketch will suffice.

Theorem 6 *In both the unique-winner and the nonunique-winner model, Baldwin-*DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof Sketch. To prove NP-hardness, we reduce the NP-complete problem X3C to Baldwin-DESTRUCTIVE-SHIFT-BRIBERY. From a given X3C instance $(X,\mathscr{S})$, we construct an election (C,V), where $C = \{p,w,b,d\} \cup X \cup \mathscr{S}$ is the set of candidates, p is the designated candidate, and V consists of two lists of votes, V_1 and V_2, where V_1 contains the following votes:

#	votes	for	#	votes	for
1	$W_{(p,S_j)}$	$1 \leq j \leq 3m$	2	$W_{(w,x_i)}$	$1 \leq i \leq 3m$
2	$W_{(S_j,x_{j,1})}$	$1 \leq j \leq 3m$	$3m+7$	$W_{(w,d)}$	
2	$W_{(S_j,x_{j,2})}$	$1 \leq j \leq 3m$	$m+10$	$W_{(b,S_j)}$	$1 \leq j \leq 3m$
2	$W_{(S_j,x_{j,3})}$	$1 \leq j \leq 3m$			

Furthermore, V_2 contains the following votes:

#	votes	for	#	votes
1	$W_{(d,p)}$		$6m+14$	$W_{(p,w)}$
$2m+7$	$W_{(p,S_j)}$	$1 \leq j \leq 3m$	$3m^2+33m+12$	$W_{(p,b)}$
$3m+3$	$W_{(p,x_i)}$	$1 \leq i \leq 3m$		

Let $V = V_1 \cup V_2$. Then we have the following Borda scores for the complete profile V:

$$
\begin{aligned}
score_{(C,V)}(x_i) &= avg(V) - 3m - 11 \text{ for every } x_i \in X,\\
score_{(C,V)}(S_j) &= avg(V) - 3m - 12 \text{ for every } S_j \in \mathscr{S},\\
score_{(C,V)}(d) &= avg(V) - 3m - 6,\\
score_{(C,V)}(w) &= avg(V) + 3m - 7,\\
score_{(C,V)}(b) &= avg(V) - 3m - 12,\\
score_{(C,V)}(p) &= avg(V) + 18m^2 + 72m + 25.
\end{aligned}
$$

Regarding the price function, for every first vote of $W_{(p,S_j)}$ (i.e., a vote of the form $p\ S_j\ C \setminus \{S_j, p\}$), let $\rho(1) = 1$ and $\rho(t) = m+1$ for every $t \geq 2$. For every other vote, let $\rho(t) = m+1$ for every $t \geq 1$. Finally, we set the budget $B = m$.

It is easy to see that p wins the election (C,V).

We claim that $(X,\mathscr{S})$ is in X3C if and only if $((C,V),p,B,\rho)$ is in Baldwin-DESTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose there is an exact cover $\mathscr{S}' \subseteq \mathscr{S}$. Then we bribe the first votes of $W_{(p,S_j)}$ for every $S_j \in \mathscr{S}'$ by shifting p to the right once. With a similar argument as in the proof of Theorem 5, d alone wins the election, i.e., p is not among the winners.

($\Leftarrow$) Suppose there is no exact cover. Then, for every $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \leq m$, there is at least one $x_i \in X$ that is not covered by $\mathscr{S}'$. It is obvious that at most m of the first votes of $W_{(p,S_j)}$ can be bribed without exceeding the budget. We can then show, similarly as in the proof of Theorem 5, that d will always be eliminated before w and therefore p cannot be prevented from winning the election alone. □

Finally, we turn to Nanson elections for which we again will show that shift bribery is NP-hard. The reduction below will only use pairs of votes of the form $W_{(c_1,c_2)}$. The average Borda score for those two votes is $|C|-1$. The candidate c_1 scores one point more than the average Borda score and c_2 scores one point fewer than the average Borda score. The other candidates score exactly the average Borda score. If a candidate is eliminated in a round, the average Borda score required to survive the next round decreases by one. Regardless of which candidate is eliminated, all remaining candidates that are not c_1 or c_2 lose one point and still have exactly the average Borda score. If c_2 is eliminated, c_1 loses its advantage with respect to the average Borda score and now scores exactly the average Borda score as well. If one of the other candidates is eliminated, c_1 continues to have one point more than the average Borda score. By symmetry, this holds analogously for c_2: If c_1 is eliminated, c_2 scores the average Borda score, and if one of the other candidates is eliminated, c_2 still has one point fewer than the average Borda score.

Theorem 7 *In both the unique-winner and the nonunique-winner model, Nanson-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce the NP-complete problem X3C to Nanson-CONSTRUCTIVE-SHIFT-BRIBERY. Again, starting from a given X3C instance $(X,\mathscr{S})$, we construct an election (C,V) with the set of candidates $C = \{p, w_1, w_2, d\} \cup X \cup \mathscr{S}$, where p is the designated candidate. Then we construct two sets of votes, V_1 and V_2, where V_1 contains the following votes:

#	votes	for	#	votes	for
1	$W_{(S_j,p)}$	$1 \leq j \leq 3m$	1	$W_{(x_{j,3},S_j)}$	$1 \leq j \leq 3m$
1	$W_{(x_i,p)}$	$1 \leq i \leq 3m$	4	$W_{(S_j,w_1)}$	$1 \leq j \leq 3m$
1	$W_{(x_{j,1},S_j)}$	$1 \leq j \leq 3m$	$15m$	$W_{(w_1,w_2)}$	
1	$W_{(x_{j,2},S_j)}$	$1 \leq j \leq 3m$	$3m$	$W_{(p,w_1)}$	

Furthermore, V_2 contains the following votes:

#	votes	for
$2m$	$W_{(p,d)}$	
2	$W_{(d,S_j)}$	$1 \leq j \leq 3m$
4	$W_{(d,x_i)}$	$1 \leq i \leq 3m$

Let $V = V_1 \cup V_2$. Then we have the following Borda scores for the complete profile V:

$$\begin{aligned} score_{(C,V)}(x_i) &= avg(V) \text{ for every } x_i \in X, \\ score_{(C,V)}(S_j) &= avg(V) \text{ for every } S_j \in \mathscr{S}, \\ score_{(C,V)}(p) &= avg(V) - m, \\ score_{(C,V)}(w_1) &= avg(V), \\ score_{(C,V)}(w_2) &= avg(V) - 15m, \\ score_{(C,V)}(d) &= avg(V) + 16m. \end{aligned}$$

The price function is again defined as follows. For every first vote of $W_{(S_j,p)}$ (i.e., a vote of the form $S_j \; p \; C \setminus \{S_j, p\}$), let $\rho(1) = 1$ and $\rho(t) = m+1$ for every $t \geq 2$. For every other vote, let $\rho(t) = m+1$ for every $t \geq 1$. Finally, we set the budget $B = m$.

It is easy to see that p is eliminated in the first round of the election (C,V) and so does not win.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Nanson-CONSTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose there is an exact cover $\mathscr{S}' \subseteq \mathscr{S}$. Then, for every $S_j \in \mathscr{S}'$, we bribe the first vote of $W_{(S_j,p)}$ by shifting p to the left once in all those votes. Note that we won't exceed our budget, since this bribe action costs 1 per vote and $|\mathscr{S}'| = m$. With the additional m points, p reaches the average Borda score and is not eliminated in the first round. However, all candidates in $\mathscr{S}'$ lose one point and are eliminated. Additionally, w_2 will be eliminated as well.

In the next round, w_1 will be eliminated, since she has $11m$ points fewer than the average Borda score required to survive this round. Since the candidates in $\mathscr{S}'$ were eliminated in the last round and $\mathscr{S}'$ is an exact cover, every candidate in X now has fewer points than the average Borda score and is eliminated.

In the third round, only p, d, and the candidates in $\mathscr{S} \setminus \mathscr{S}'$ are still standing. Therefore, the only pairs of votes that are not symmetric are $W_{(S_j,p)}$, twice $W_{(d,S_j)}$ for every $S_j \in (\mathscr{S} \setminus \mathscr{S}')$, and $2m$ pairs of $W_{(p,d)}$. Since $|\mathscr{S} \setminus \mathscr{S}'| = 2m$, we have that p scores exactly the average Borda score and survives this round, just as d. Every $S_j \in (\mathscr{S} \setminus \mathscr{S}')$ has one point fewer than the average Borda score and is eliminated. This leaves only p and d in the last round, which p alone wins.

($\Leftarrow$) Suppose there is no exact cover. Then, for every $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| = m$, there is at least one $x_i \in X$ that is not covered by $\mathscr{S}'$. Note that we can only bribe the first votes of any $W_{(S_j,p)}$ without exceeding the budget. For p to survive the first round, we need to bribe m of those votes by shifting p to the left once. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that $\mathscr{S}'$ contains S_j exactly if the first vote of $W_{(S_j,p)}$ has been bribed. Then every $S_j \in \mathscr{S}'$ has a score of $avg(V) - 1$ and p has a score of $avg(V)$. Therefore, in the first round, every candidate from $\mathscr{S}'$ and w_2 are eliminated from the election.

In the second round, w_1 will be eliminated because of the $15m$ pairs of votes $W_{(w_1,w_2)}$ and the elimination of w_2. Furthermore, a candidate $x_i \in X$ reaches the average Borda score with p and d still standing only if all three $S_j \in \mathscr{S}$ with $x_i \in S_j$ are also not yet eliminated. Since the candidates in $\mathscr{S}'$ were eliminated in the previous round, for every $S_j \in \mathscr{S}'$, all three $x_i \in S_j$ will be eliminated in this round. Since $\mathscr{S}'$ is not an exact cover, there are candidates $X' \subseteq X$ that survive this round. d also reaches the average Borda score, as there are $2m$ candidates $\mathscr{S} \setminus \mathscr{S}'$ and those candidates $\mathscr{S} \setminus \mathscr{S}'$ survive due to w_1.

In the next round, the candidates still standing are p, d, X', and $\mathscr{S} \setminus \mathscr{S}'$. Because $|X'| \geq 1$, candidate p has $|X'|$ points fewer than the average Borda score and is eliminated in this round. Thus p does not win. □

Our last result in this section shows that the destructive variant of shift bribery in Nanson elections is intractable as well.

Theorem 8 *In both the unique-winner and the nonunique-winner model, Nanson-*DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce the NP-complete problem X3C to Nanson-DESTRUCTIVE-SHIFT-BRIBERY. Once more, given an X3C instance $(X, \mathscr{S})$, we construct an election (C,V) with the set of candidates $C = \{p, w_1, w_2, w_3, d\} \cup X \cup \mathscr{S}$, where p is the designated candidate and $(X, \mathscr{S})$ is the given X3C instance. Then we construct two sets of votes, V_1 and V_2, where V_1 contains the following votes:

#	votes	for	#	votes	for
1	$W_{(p,S_j)}$	$1 \le j \le 3m$	6	$W_{(S_j,w_3)}$	$1 \le j \le 3m$
1	$W_{(d,x_i)}$	$1 \le i \le 3m$	$20m$	$W_{(w_1,w_2)}$	
2	$W_{(x_{j,1},S_j)}$	$1 \le j \le 3m$	$19m$	$W_{(w_3,w_1)}$	
2	$W_{(x_{j,2},S_j)}$	$1 \le j \le 3m$	$3m+1$	$W_{(w_3,d)}$	
2	$W_{(x_{j,3},S_j)}$	$1 \le j \le 3m$			

Furthermore, V_2 contains the following votes:

#	votes	for	#	votes
1	$W_{(d,p)}$		$3m+1$	$W_{(p,w_3)}$
1	$W_{(p,x_i)}$	$1 \le i \le 3m$		

Let $V = V_1 \cup V_2$. Then we have the following Borda scores for the complete profile V:

$$\begin{aligned}
score_{(C,V)}(x_i) &= avg(V) + 4 \text{ for every } x_i \in X,\\
score_{(C,V)}(S_j) &= avg(V) - 1 \text{ for every } S_j \in \mathscr{S},\\
score_{(C,V)}(d) &= avg(V),\\
score_{(C,V)}(w_1) &= avg(V) + m,\\
score_{(C,V)}(w_2) &= avg(V) - 20m,\\
score_{(C,V)}(w_3) &= avg(V) + m,\\
score_{(C,V)}(p) &= avg(V) + 9m.
\end{aligned}$$

The price function is again defined as follows. For every first vote of $W_{(p,S_j)}$ (i.e., a vote of the form $p\ S_j\ C \setminus \{S_j, p\}$), let $\rho(1) = 1$ and $\rho(t) = m+1$ for every $t \ge 2$. For every other vote, let $\rho(t) = m+1$ for every $t \ge 1$. Finally, we set the budget $B = m$.

It is easy to see that p will only have fewer points than the average Borda score if all candidates from $\mathscr{S}$, X, and the candidate w_3 are eliminated while d is still standing. Without bribing, d is eliminated in the third round while w_3 is still standing, and eventually p wins the election (C,V).

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Nanson-DESTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model.

($\Rightarrow$) Suppose there is an exact cover $\mathscr{S}' \subseteq \mathscr{S}$. Then, for every $S_j \in \mathscr{S}'$, we bribe the first vote of $W_{(p,S_j)}$ by shifting p to the right once in all those votes. Note that we won't exceed our budget, since this bribe action costs 1 per vote and $|S'| = m$. After those m votes have been bribed, every $S_j \in \mathscr{S}'$ gains a point and therefore survives the first round. All other candidates $\mathscr{S} \setminus \mathscr{S}'$ and w_2 are eliminated.

Let $C_1 = \{p, d, w_1, w_3\} \cup X \cup \mathscr{S}'$ be the set of candidates present in the second round. w_1 loses $19m$ points on the average Borda score from the elimination of w_2 and is eliminated. Additionally, all candidates of X lose four points on the average Borda score but still survive this round, as they now have exactly the average Borda score.

Let $C_2 = \{p, d, w_3\} \cup X \cup \mathscr{S}'$ be the candidates in the third round. In this round, only w_3 is eliminated because w_3 lost $19m$ points on the average Borda score from the elimination of w_1.

Let $C_3 = \{p, d\} \cup X \cup \mathscr{S}'$ be the candidates in the fourth round. The scores are as follows:

$$\begin{aligned} score_{(C_3,V)}(x_i) &= avg(V) \text{ for every } x_i \in X, \\ score_{(C_3,V)}(S_j) &= avg(V) - 6 \text{ for every } S_j \in \mathscr{S}', \\ score_{(C_3,V)}(d) &= avg(V) + 3m + 1, \\ score_{(C_3,V)}(p) &= avg(V) + 3m - 1. \end{aligned}$$

Therefore all candidates in $\mathscr{S}'$ are eliminated. In the following round, all candidates in X are eliminated. This leaves only p and d in the final round in which p is eliminated and thus cannot win.

($\Leftarrow$) Suppose there is no exact cover. Then, for every $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \leq m$, there is at least one $x_i \in X$ that is not covered by $\mathscr{S}'$. Note that we can only bribe the first votes of any $W_{(p,S_j)}$ without exceeding the budget.

We now show that, even with optimal bribing, d will be eliminated in the third round and, therefore, p alone wins the election. Within our budget, we can prevent at most m candidates from $\mathscr{S}$, say $\mathscr{S}'$, of being eliminated in the first round by bribing the corresponding vote of $W_{(p,S_j)}$. Since $\mathscr{S}'$ cannot be an exact cover of X, there is at least one $x_i \in X$ for which all $S_j \in \mathscr{S}$ with $x_i \in S_j$ are eliminated. This x_i is eliminated in the second round, as it has lost six points on the average Borda score from the eliminations of candidates in the previous round. In the third round, w_3 is still participating since w_2 and w_1 were only eliminated in the first and second round, respectively. Therefore, the score of d minus the average Borda score of this round is at most -1, which means that d is eliminated in this round. Thus, there is no candidate left that can prevent p from winning the election. □

5 Iterated Plurality and Plurality with Runoff

In this section, we show hardness of shift bribery for iterated plurality and plurality with runoff, handling both voting systems simultaneously and starting with the constructive case.

Theorem 9 *In both the unique-winner and the nonunique-winner model, for iterated plurality and plurality with runoff,* CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce X3C to CONSTRUCTIVE-SHIFT-BRIBERY for these two voting systems. Let $(X, \mathscr{S})$ be a given X3C instance. We construct the CONSTRUCTIVE-SHIFT-BRIBERY instance $((C,V), p, B, \rho)$ as follows. Let $C = \{p, w\} \cup X \cup \mathscr{S} \cup D$ be the set of candidates, where p is the designated candidate and $D = \{d_{i,j} \mid 1 \leq i \leq 3m \text{ and } 1 \leq j \leq m-7\}$ is a set of dummy candidates. The list V of votes is constructed as follows:

#	vote	for
1	$S_i \; p \; \cdots$	$1 \leq i \leq 3m$
2	$S_i \; x_{i,1} \; \overrightarrow{X \setminus \{x_{i,1}\}} \cdots$	$1 \leq i \leq 3m$
2	$S_i \; x_{i,2} \; \overrightarrow{X \setminus \{x_{i,2}\}} \cdots$	$1 \leq i \leq 3m$
2	$S_i \; x_{i,3} \; \overrightarrow{X \setminus \{x_{i,3}\}} \; \cdots$	$1 \leq i \leq 3m$
1	$S_i \; d_{i,j} \; \overrightarrow{X \setminus \{x_i\}} \; \cdots$	$1 \leq i \leq 3m, \; 1 \leq j \leq m-7$
m	$x_i \; \overrightarrow{X \setminus \{x_i\}} \; \cdots$	$1 \leq i \leq 3m$
m	$d_{i,j} \; \overrightarrow{X} \; \cdots$	$1 \leq i \leq 3m, \; 1 \leq j \leq m-7$
3	$w \; p \; \cdots$	

For voters with votes of the form $S_i\ p\ \cdots$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \geq 2$; and for every other voter, we use the price function $\rho(t) = m+1$ for $t \geq 1$. Finally, set the budget $B = m$.

Without bribing, p has a score of zero and is eliminated immediately in both voting systems.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in CONSTRUCTIVE-SHIFT-BRIBERY for either of the two voting systems, regardless of the winner model.

($\Rightarrow$) Suppose that $(X, \mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. We now show that it is possible for p to become a unique iterated-plurality (respectively, plurality-with-runoff) winner of an election obtained by shifting p in the votes without exceeding the budget. For every $S_i \in \mathscr{S}'$, we bribe the voter with the vote of the form $S_i\ p\ \cdots$, so her new vote is of the form $p\ S_i\ \cdots$. In the first round p, every $x_i \in X$, every $d_{i,j} \in D$, and every $S_i \in \mathscr{S} \setminus \mathscr{S}'$ is a plurality winner, so only these candidates participate in the next round. In the second round, p receives three further points from the three voters whose vote is $w\ p\ \cdots$. Every candidate $x_j \in X$ receives two further points from the votes of the form $S_i\ x_j\ \cdots$ with $x_j \in S_i$ and $S_i \in \mathscr{S}'$. Every $d_{i,j}$ with $S_i \in \mathscr{S}'$ and $1 \leq j \leq m-7$ receives one additional point from the voters with vote $S_i\ d_{i,j}\ \cdots$. It follows that p has the most points and therefore p is the unique iterated-plurality (respectively, plurality-with-runoff) winner.

($\Leftarrow$) Suppose that $(X, \mathscr{S})$ is a no-instance of X3C. Then, for every $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| = m$, there is at least one candidate in X that is not covered and, therefore, at least one candidate in X occurring in at least two sets from $\mathscr{S}'$. We show that it is not possible for p to become a winner of the election obtained from the original election by bribing without exceeding the budget.

To become a winner of such a bribed election, it is necessary for p to get at least m points in the first round. Due to the budget, it is also necessary to bribe m voters with a vote of the form $S_i\ p\ \cdots$ with $S_i \in \mathscr{S}'$. It follows that p, each $x \in X$, each $S_i \in \mathscr{S} \setminus \mathscr{S}'$, and each $d_{i,j} \in D$ participate in the second round. As mentioned above, at least one candidate in X receives at least four further points due to the fact that $\mathscr{S}'$ is not a cover of X. Thus p does not win. That means that $((C,V), p, B, \rho)$ is a no-instance of CONSTRUCTIVE-SHIFT-BRIBERY for either of iterated plurality and plurality with runoff regardless of the winner model. □

We have the same result in the destructive case. This is the first proof where we use an NP-complete problem other than X3C to show NP-hardness, namely ONE-IN-THREE-POSITIVE-3SAT, which was also defined in Section 2.

Theorem 10 *In both the unique-winner and the nonunique-winner model, for iterated plurality and plurality with runoff,* DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce the NP-complete problem ONE-IN-THREE-POSITIVE-3SAT to DESTRUCTIVE-SHIFT-BRIBERY for both voting systems. Let (X, S) be a given ONE-IN-THREE-POSITIVE-3SAT instance, where $X = \{x_1, \ldots, x_{3m}\}$ and $S = \{S_1, \ldots S_{3m}\}$ with $S_i = \{x_{i,1}, x_{i,2}, x_{i,3}\} \subseteq X$ for each $1 \leq i \leq 3m$. Without loss of generality, we can assume that $m > 6$. We construct the DESTRUCTIVE-SHIFT-BRIBERY instance for both voting systems as follows. Let $C = \{p, w, e, f\} \cup D \cup Y \cup X$ with $D = \{d_{i,j} \mid 1 \leq i \leq 3m \text{ and } 1 \leq j \leq 2m-1\}$ and $Y = \{y_{i,j} \mid 1 \leq i \leq 3m \text{ and } 1 \leq j \leq 4\}$ and where p is the designated candidate. The list V of votes is constructed as follows:

#	votes	for
1	$p\ x_i\ \cdots$	$1 \leq i \leq 3m$
1	$y_{i,1}\ x_{i,1}\ x_{i,2}\ w\ p\ \cdots$	$1 \leq i \leq 3m$
1	$y_{i,2}\ x_{i,2}\ x_{i,3}\ w\ p\ \cdots$	$1 \leq i \leq 3m$
1	$y_{i,3}\ x_{i,1}\ x_{i,3}\ w\ p\ \cdots$	$1 \leq i \leq 3m$
4	$y_{i,4}\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p\ \cdots$	$1 \leq i \leq 3m$
1	$x_i\ d_{i,j}\ p\ \cdots$	$1 \leq i \leq 3m$, $1 \leq j \leq 2m-1$
$2m$	$d_{i,j}\ p\ \cdots$	$1 \leq i \leq 3m$, $1 \leq j \leq 2m-1$
$2m$	$w\ p\ \cdots$	
$2m-1$	$e\ p\ \cdots$	
m	$f\ p\ \cdots$	

For votes of the form $p\ x_i\ \cdots$ we use the price function $\rho(1) = 1$ and $p(t) = m+1$ for all $t \geq 2$. For every other vote, we use the price function $\rho(t) = m+1$ for $t \geq 1$. Finally, set the budget $B = m$.

Without bribing, the election proceeds as follows. In the first round, p scores $3m$ points, w and every $d_{i,j} \in D$ scores $2m$ points, and each of the remaining candidates scores fewer than $2m$ points. In the second round, p scores $18m-1$ points, w scores $11m$ points, and every $d_{i,j}$ scores $2m+1$ points. It follows that p is the unique winner for either of iterated plurality and plurality with runoff.

We claim that (X,S) is in ONE-IN-THREE-POSITIVE-3SAT if and only if $((C,V),p,B,\rho)$ is in DESTRUCTIVE-SHIFT-BRIBERY for either of the two voting systems, regardless of the winner model.

($\Rightarrow$) Suppose that (X,S) is a yes-instance of ONE-IN-THREE-POSITIVE-3SAT. Then there exists a subset $U \subseteq X$ such that for each clause S_j we have $|U \cap S_j| = 1$. We bribe the voters with the vote of the form $p\ x_i\ \cdots$ with $x_i \in U$ so that the new vote has the form $x_i\ p\ \cdots$. It follows that p, w, every $x_i \in U$, and every $d_{i,j} \in D$ reach the second round with $2m$ points each. In the second round, p gains $3m-1$ additional points while w gains $3m$ additional points. It follows that p is not a winner of the election, so $((C,V),p,B,\rho)$ is a yes-instance of DESTRUCTIVE-SHIFT-BRIBERY for both voting systems, regardless of the winner model.

($\Leftarrow$) Suppose that (X,S) is a no-instance of ONE-IN-THREE-POSITIVE-3SAT. We show that $((C,V),p,B,\rho)$ is also a no-instance of DESTRUCTIVE-SHIFT-BRIBERY for both voting systems. To ensure that p is not the only plurality winner in the first round, it is necessary to bribe m voters with votes of the form $p\ x_i\ \cdots$ to now vote $x_i\ p\ \cdots$. Note that we can only bribe at most m such voters without exceeding the budget. Let $U \subseteq X$ be the set of candidates that benefit from the bribery action. It follows that p, every $d_{i,j} \in D$, every $x_i \in U$, and w can move forward to the next round with $2m$ points each. In this round, the designated candidate p gains $3m-1$ additional points from the votes of the form $e\ p\ \cdots$ and $f\ p\ \cdots$; every candidate $d_{i,j}$ with $x_i \notin U$ gains one additional point; every candidate $x_i \in U$ can receive at most 18 additional points; and w is discussed separately in the following paragraph.

To prevent the victory of p, it is necessary that w gains at least $3m$ points (since if w gains only $3m-1$ points, it follows that w and p move forward to the final round, where p would achieve a clear victory). For w to gain at least one point from any one of the three votes of the form $y_{i,1}\ x_{i,1}\ x_{i,2}\ w\ p\ \cdots$, $y_{i,2}\ x_{i,2}\ x_{i,3}\ w\ p\ \cdots$, and $y_{i,3}\ x_{i,1}\ x_{i,3}\ w\ p\ \cdots$, it is necessary that at most one candidate $x_{i,j}$ participates in the second round. On the other hand, if no candidate $x_{i,j}$ participates in the second round, p gains four points from the voters of the fifth line, whose vote is $y_{i,4}\ x_{i,1}\ x_{i,2}\ x_{i,3}\ p\ \cdots$, i.e., this clause harms w. Only a clause S_i with $|S_i \cap U| = 1$ helps w to reduce the point difference to p. Since (X,S) is a no-instance of ONE-IN-THREE-POSITIVE-3SAT, there are at most $3m-2$ clauses with this property.

With these clauses w can reduce the point difference to two. With the two remaining clauses the point difference is growing. This implies that p is always a unique winner of the election, i.e., $((C,V),p,B,\rho)$ is a no-instance of DESTRUCTIVE-SHIFT-BRIBERY for both voting systems, regardless of the winner model. □

6 Iterated Veto and Veto with Runoff

In this section, we show hardness of shift bribery for iterated veto and veto with runoff, again handling both voting systems simultaneously and starting with the constructive case.

Theorem 11 *In both the unique-winner and the nonunique-winner model, for veto with runoff and iterated veto,* CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce X3C to CONSTRUCTIVE-SHIFT-BRIBERY for veto with runoff and iterated veto at the same time. Let $(X,\mathscr{S})$ be a given X3C instance and construct the CONSTRUCTIVE-SHIFT-BRIBERY instance $((C,V),p,B,\rho)$ as follows. Let $C = \{p,d_1,d_2\} \cup X \cup \mathscr{S}$ be the set of candidates, where p is the designated candidate, and construct the voter preferences in V as follows:

#	votes	for
1	$\cdots\ S_i\ p$	$1 \le i \le 3m$
2	$\cdots\ x_{i,1}\ S_i$	$1 \le i \le 3m$
2	$\cdots\ x_{i,2}\ S_i$	$1 \le i \le 3m$
2	$\cdots\ x_{i,3}\ S_i$	$1 \le i \le 3m$
$2m-6$	$\cdots\ d_2\ S_i$	$1 \le i \le 3m$
$2m$	$\cdots\ x_i$	$1 \le i \le 3m$
m	$\cdots\ d_2\ x_i\ d_1$	$1 \le i \le 3m$
$m+2$	$\cdots\ d_2\ S_i\ d_1$	$1 \le i \le 3m$
$2m$	$\cdots\ d_2$	
1	$\cdots\ p\ d_1$	

For votes of the form $\cdots\ S_i\ p$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \ge 2$; and for every other voter, we use the price function $\rho(t) = m+1$ for $t \ge 1$. Finally, set the budget $B = m$.

Note that for both voting rules, p is eliminated in the first round with $3m$ vetoes and therefore cannot be the winner without bribing voters.

We claim that $(X,\mathscr{S})$ is in X3C if and only if $((C,V),p,B,\rho)$ is in CONSTRUCTIVE-SHIFT-BRIBERY for either of iterated veto and veto with runoff, regardless of the winner model.

($\Rightarrow$) Suppose that $(X,\mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. Shift p one position forward in the votes of the form $\cdots\ S_i\ p$ for each $S_i \in \mathscr{S}'$, so that the new vote has the form $\cdots\ p\ S_i$. It follows that p, each $S \in \mathscr{S} \setminus \mathscr{S}'$, each x_i for $1 \le i \le 3m$, and d_2 are veto winners with $2m$ vetoes each and thus proceed to the second round. Since $\mathscr{S}'$ is an exact cover, each x_i receives two additional vetoes from the voters in lines 2–4 corresponding to the sets in the exact cover and m vetoes from the voters in line 7. Furthermore, each $S \in \mathscr{S} \setminus \mathscr{S}'$ receives $m+2$ vetoes from the voters in line 8, whereas p receives m vetoes from the voters in line 1 and only one additional veto from the voter in the last line. Since d_2 gains far more than $m+1$ vetoes in this round, it follows that p is the unique veto winner of the bribed election. Thus $((C,V),p,B,\rho)$ is a yes-instance of CONSTRUCTIVE-SHIFT-BRIBERY for either of iterated veto and veto with runoff, regardless of the winner model.

($\Leftarrow$) Suppose that $(X,\mathscr{S})$ is a no-instance of X3C. This means that for every $\mathscr{S}' \subseteq \mathscr{S}$, $|\mathscr{S}'| \le m$, there is an $x' \in X$ that is not covered by any $S \in \mathscr{S}'$.

To not be eliminated in the first round and to not exceed the budget of m, p has to lose exactly m vetoes so as to tie with the $2m$ vetoes of the x_i. This is only possible by bribing the voters in the first line. Let $\mathscr{S}' \subseteq \mathscr{S}$, $|\mathscr{S}'| = m$, be the set that corresponds to the S_i of the bribed voters. Candidates p and d_2 as well as each $S \in \mathscr{S} \setminus \mathscr{S}'$ and each x_i, $1 \le i \le 3m$, reach the second round with $2m$ vetoes. However, in the second round, the $x' \in X$ that was not covered by $\mathscr{S}'$ receives only m additional vetoes in contrast to p who receives $m+1$ additional vetoes. It

follows that p is not winning the election for either of the two voting rules. That means that $((C,V),p,B,\rho)$ is a no-instance of CONSTRUCTIVE-SHIFT-BRIBERY for either of iterated veto and veto with runoff, regardless of the winner model. □

We now turn to the destructive variant of shift bribery for iterated veto and veto with runoff.

Theorem 12 *In both the unique-winner and the nonunique-winner model, for veto with runoff and iterated veto,* DESTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard.*

Proof. To prove NP-hardness, we reduce the NP-complete problem ONE-IN-THREE-POSITIVE-3SAT to DESTRUCTIVE-SHIFT-BRIBERY for veto with runoff and iterated veto simultaneously. Given an instance (X,S) of ONE-IN-THREE-POSITIVE-3SAT, where $X = \{x_1,\dots,x_{3m}\}$ and $S = \{S_1,\dots,S_{3m}\}$, with $S_i = \{x_{i,1},x_{i,2},x_{i,3}\} \subseteq X$ for each $1 \leq i \leq 3m$, we construct the election (C,V) with candidate set $C = \{p,w,d_1,d_2\} \cup X$, designated candidate p, and the following list V of votes:

#	votes	for
1	$\cdots\ p\ x_i$	$1 \leq i \leq 3m$
2	$\cdots\ p\ x_{i,1}\ x_{i,2}\ d_1$	$1 \leq i \leq 3m$
2	$\cdots\ p\ x_{i,2}\ x_{i,3}\ d_1$	$1 \leq i \leq 3m$
2	$\cdots\ p\ x_{i,1}\ x_{i,3}\ d_1$	$1 \leq i \leq 3m$
7	$\cdots\ w\ x_{i,1}\ x_{i,2}\ x_{i,3}\ d_1$	$1 \leq i \leq 3m$
$2m$	$\cdots\ d_2\ x_i$	$1 \leq i \leq 3m$
$22m$	$\cdots\ d_2\ x_i\ d_1$	$1 \leq i \leq 3m$
$2m$	$\cdots\ d_2$	
m	$\cdots\ p$	
$2m$	$\cdots\ w$	
$8m-1$	$\cdots\ w\ d_1$	

For every vote of the form $\cdots\ p\ x_i$, let the price function be $\rho(1) = 1$ and $\rho(t) = m+1$ for every $t \geq 2$. For every other vote, define $\rho(t) = m+1$ for every $t \geq 1$. Finally, we set the budget $B = m$.

It is easy to see that p is the winner of this election for both voting rules.

We claim that (X,S) is in ONE-IN-THREE-POSITIVE-3SAT if and only if $((C,V),p,B,\rho)$ is in DESTRUCTIVE-SHIFT-BRIBERY for either of veto with runoff and iterated veto, regardless of the winner model.

($\Rightarrow$) Assume that (X,S) is in ONE-IN-THREE-POSITIVE-3SAT. Then there is a subset $X' \subseteq X$ such that for each clause S_i we have $|X' \cap S_i| = 1$. Bribe the voters with votes of the form $\cdots\ p\ x_i$ with $x_i \in X'$ so that the new vote has the form $\cdots\ x_i\ p$. It follows that p, w, d_2, and each $x_i \in X'$ have the fewest vetoes (namely, $2m$) and therefore proceed to the second round. In the second round, p receives $2m$ vetoes from the votes in line 1 and for each of the $3m$ clauses two vetoes from the voters in lines 2–4 for a total of $8m$ additional vetoes, whereas w only receives a total of $8m-1$ vetoes. It follows that p is not a winner of the election for either of the two voting rules.

($\Leftarrow$) Let (X,S) be a yes-instance of DESTRUCTIVE-SHIFT-BRIBERY for veto with runoff (respectively, iterated veto), i.e., it is possible to bribe voters so that p does not win the election. Recall that it is only possible to bribe voters in line 1 without ecxeeding the budget. In the first round, p receives m vetoes, i.e., the fewest vetoes of all candidates. Due to the votes in line 7, the only candidate capable of receiving fewer vetoes than p or the same number of vetoes as p in the second round is w.[9] However, this is only possible if p receives at least $9m-1$ additional vetoes since w has $10m-1$ vetoes in the second round from the last two lines alone. p receives $3m$ of these additional vetoes from line 1—after bribing voters so that p is in the last position, or eliminating the x_i in the first round—leaving a gap of

[9] Note that d_1 will definitely be eliminated in the first round.

$6m-1$ vetoes. For each clause S_j such that no $x_i \in S_j$ is present in the second round, p receives six additional vetoes (lines 2–4), whereas w receives in this case seven additional vetoes from the voters in line 5, i.e., this widens the gap between p and w instead of closing it. That means that for each clause S_j, there has to be at least one $x_i \in S_j$ present in the second round, i.e., for each clause S_j, a voter with a vote of the form $\cdots\ p\ x_i$ with $x_i \in S_j$ needs to be bribed to cast a vote of the form $\cdots\ x_i\ p$ to bring the respective vetoes down to $2m$, the same as, e.g., d_2. However, if at least two literals, say x_i and x_k, in a clause S_j are present in the second round, p receives no additional veto, which does not help to close the gap between p and w. The only possibility remaining for p not to be a winner of the bribed election is that the bribed voters correspond to the variables set to true in an assignment where in each clause there is exactly one literal true, i.e., we have a yes-instance of ONE-IN-THREE-POSITIVE-3SAT. □

7 Using the Nonmonotonicity Property

Informally stated, a voting rule is said to be *monotonic* if winners can never be turned into nonwinners by improving their position in some votes, everything else remaining the same.[10] Intuitively, that is to say that only shifting a candidate forward (closer to the top) is beneficial, whereas shifting a candidate backward (closer to the bottom) is not. In shift bribery under some monotonic voting rule, it thus makes only sense for the briber to shift the designated candidate forward in the constructive case (respectively, backward in the destructive case). However, all voting rules considered here except iterated plurality and iterated veto are *not* monotonic, and in nonmonotonic voting rules, shifting the designated candidate backward in the constructive case (respectively, forward in the destructive case) could also be beneficial for the briber.

It would therefore be interesting to find out whether the complexity of our problems changes when the nonmonotonicity of voting rules is specifically allowed, or even required, to be exploited in shift bribery actions. Indeed, with slight modifications to the proofs, we can show that Hare-CONSTRUCTIVE-SHIFT-BRIBERY and plurality-with-runoff-CONSTRUCTIVE-SHIFT-BRIBERY are still NP-hard if the designated candidate can *only* be shifted *backward*. We conjecture that all other proofs (except the proofs for the monotonic voting rules iterated plurality and iterated veto) can be adapted in such a way as well.

We start with constructive shift bribery in Hare elections where the only allowed bribery action is to shift the designated candidate *backward*.

Theorem 13 *In both the unique-winner and the nonunique-winner model, Hare-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard even if the designated candidate can only be shifted backward.*

Proof. NP-hardness again follows by a reduction from X3C. Construct from a given X3C instance $(X, \mathscr{S})$ an instance $((C,V), p, B, \rho)$ of Hare-CONSTRUCTIVE-SHIFT-BRIBERY with candidate set $C = X \cup \mathscr{S} \cup D \cup \{p, w\}$, where $D = \{d_1, \ldots, d_{3m}\}$ is a set of dummy candidates and p the designated candidate, and the following list V of votes:

[10] This definition captures just one common notion of monotonicity, the one we will be using here; but note that there are also other notions of monotonicity for voting rules known in social choice theory.

#	vote	for
1	$S_i\ x_{i,1}\ \overrightarrow{X \setminus \{x_{i,1}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
1	$S_i\ x_{i,2}\ \overrightarrow{X \setminus \{x_{i,2}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
1	$S_i\ x_{i,3}\ \overrightarrow{X \setminus \{x_{i,3}\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
4	$x_i\ \overrightarrow{X \setminus \{x_i\}}\ w\ p\ \cdots$	$1 \le i \le 3m$
6	$w\ \overrightarrow{X}\ p\ \cdots$	
1	$p\ S_i\ \cdots$	$1 \le i \le 3m$
6	$p\ \cdots$	
3	$d_i\ S_i\ p\ w \cdots$	$1 \le i \le 3m$

For votes of the form $p\ S_i\ \cdots$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \ge 2$; and for every other vote, we use the price function ρ with $\rho(t) = m+1$ for all $t \ge 1$. Finally, set the budget $B = m$.

Without bribing the voters the election proceeds as follows:

Round	p	w	x_1	$x_i \in X \setminus \{x_1\}$	$S_i \in \mathscr{S}$	$d_i \in D$
1	$3m+6$	6	4	4	3	3
2	$12m+6$	6	7	7	out	out
3	$12m+6$	out	13	7	out	out
4	$12m+6$	out	$21m+6$	out	out	out

It follows that p is eliminated in the last round and does not win the election.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in Hare-CONSTRUCTIVE-SHIFT-BRIBERY, regardless of the winner model, even if the designated candidate can only be shifted backward.

($\Rightarrow$) Suppose that $(X, \mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. We now show that it is possible for p to become a unique Hare winner of an election obtained by shifting p in the votes without exceeding the budget B. For every $S_i \in \mathscr{S}'$, we bribe the voter with the vote of the form $p\ S_i\ \cdots$ by shifting p once, so her new vote is of the form $S_i\ p\ \cdots$; each such bribe action costs us only 1 from our budget, so the budget will not be exceeded. Now the election proceeds as follows:

Round	p	w	$x_i \in X$	$S_i \in \mathscr{S}'$	$S_i \in \mathscr{S} \setminus \mathscr{S}'$	$d_i \in D$
1	$2m+6$	6	4	4	3	3
2	$8m+6$	6	6	7	out	out
3	$26m+12$	out	out	7	out	out

We see that p is the only candidate still standing in the fourth round and thus the only Hare winner of the bribed election.

($\Leftarrow$) Suppose that $(X, \mathscr{S})$ is a no-instance of X3C. Then no subset $\mathscr{S}' \subseteq \mathscr{S}$ with $|\mathscr{S}'| \le m$ covers X. We now show that p will be eliminated in all elections obtained by bribing voters without exceeding budget B. Note that we can only bribe at most m voters with votes of the form $p\ S_i\ \cdots$ without exceeding the budget. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that for every $S_i \in \mathscr{S}'$ we have bribed the voter whose vote is $p\ S_i\ \cdots$. We can assume that $|\mathscr{S}'| > 0$.

Every candidate in $\mathscr{S}'$ will gain an additional point and therefore survives the first round. All candidates from D and $\mathscr{S} \setminus \mathscr{S}'$ will be eliminated, since p only loses at most m points.

In the second round, the remaining candidates from $\mathscr{S}$ will gain three additional points from the elimination of candidates in D and score seven points in this round (and in all subsequent rounds with p still standing). If a candidate

$S_i \in \mathscr{S}$ was eliminated in the previous round, every $x_j \in S_i$ gains one additional point in this round. Partition X into sets X_0, X_1, X_2, and X_3 so that $x_i \in X_k \Leftrightarrow |\{S_j \in \mathscr{S}' \,|\, x_i \in S_j\}| = k$ for $k \in \{0,1,2,3\}$. Note that X_0, X_1, X_2, and X_3 are disjoint and $|X_0| > 0$, but one or two of X_1, X_2, and X_3 may be empty. Then $x_i \in X_j$ scores $4+(3-j) \in \{4,5,6,7\}$ points depending on how many times x_i is covered by $\mathscr{S}'$. Therefore, every $x_i \in X_0$ scores more points than w who has six points. So, there are candidates from X that survive this round and other candidates from X (i.e., candidates from X_1, X_2, or X_3), who are eliminated.

In the third round, the candidate $x_\ell \in X$ with the smallest subscript who is still standing gains at least four points from the eliminated candidates, so that she scores at least nine points now (since no candidates from X_3 are left in the election). All other candidates still score the same number of points as in the previous round. Therefore, p scores $4|\mathscr{S} \setminus \mathscr{S}'| + 6$ points, w scores six points (if w was not already eliminated along with the candidates from X_1), every $S_i \in \mathscr{S}'$ scores seven points, and every still standing candidate from X except x_ℓ scores at most seven points. Since w can only gain additional points when all candidates from X are eliminated and only x_ℓ gains points from the elimination of w or candidates from $X \setminus \{x_\ell\}$ in the subsequent rounds, all candidates $X \setminus (\{x_\ell\} \cup X_0)$ and w are eliminated. Then all still standing candidates from $X_0 \setminus \{x_\ell\}$ and candidates from $\mathscr{S}'$ who score seven points each are eliminated, which leaves p and x_ℓ in the last round. In this round, p scores $12m+6$ points and x_ℓ scores $21m+6$ points, so p is eliminated from the election and does not win. □

Next, we show the corresponding result for plurality with runoff.

Theorem 14 *In both the unique-winner and the nonunique-winner model, plurality-with-runoff-*CONSTRUCTIVE-SHIFT-BRIBERY *is* NP*-hard even if the designated candidate can only be shifted backward.*

Proof. To prove NP-hardness, we reduce X3C to CONSTRUCTIVE-SHIFT-BRIBERY for plurality with runoff. Let $(X, \mathscr{S})$ be a given X3C instance, where $X = \{x_1, \ldots, x_{3m}\}$ and $\mathscr{S} = \{S_1, \ldots, S_{3m}\}$. Also, we require that $m > 3$. We construct the CONSTRUCTIVE-SHIFT-BRIBERY instance $((C,V), p, B, \rho)$ as follows. Let $C = \{p\} \cup X \cup \mathscr{S} \cup D \cup Y$ with sets of dummy candidates $D = \{d_{i,j} \mid 1 \le i \le 3m \text{ and } 1 \le j \le 2m^2 - 5m - 4\}$ and $Y = \{y_i \mid 1 \le i \le 3m+1\}$ and designated candidate p. The list V of votes is constructed as follows:

#	vote	for
1	$p\ S_i\ \cdots$	$1 \le i \le 3m$
2	$S_i\ x_{i,1}\ w\ \overrightarrow{X \setminus \{x_{i,1}\}}\ \cdots$	$1 \le i \le 3m$
2	$S_i\ x_{i,2}\ w\ \overrightarrow{X \setminus \{x_{i,2}\}}\ \cdots$	$1 \le i \le 3m$
2	$S_i\ x_{i,3}\ w\ \overrightarrow{X \setminus \{x_{i,3}\}}\ \cdots$	$1 \le i \le 3m$
$3m$	$w\ p\ \cdots$	
1	$y_i\ p$	$1 \le i \le 3m+1$
$m-3$	$S_i\ w\ p$	$1 \le i \le 3m$
$m-4$	$S_i\ p\ w$	$1 \le i \le 3m$
$2m$	$x_i\ w\ p$	$1 \le i \le 3m$
1	$d_{i,j}\ x_i\ w\ p\ \cdots$	$1 \le i \le 3m$, $1 \le j \le 2m^2 - 5m - 4$

For votes of the form $p\ S_i\ \cdots$, we use the price function $\rho(1) = 1$, and $\rho(t) = m+1$ for all $t \ge 2$; and for every other vote, we use the price function $\rho(t) = m+1$ for $t \ge 1$. Finally, set the budget $B = m$.

Without bribing, only p and w reach the second and final round with $3m$ points each. Clearly, w alone wins the election with only p and w present.

We claim that $(X, \mathscr{S})$ is in X3C if and only if $((C,V), p, B, \rho)$ is in CONSTRUCTIVE-SHIFT-BRIBERY for plurality with runoff, regardless of the winner model.

($\Rightarrow$) Suppose that $(X, \mathscr{S})$ is a yes-instance of X3C. Then there exists an exact cover $\mathscr{S}' \subseteq \mathscr{S}$ of size m. We now show that it is possible for p to become a unique plurality-with-runoff winner of an election obtained by shifting p in the votes without exceeding the budget. For every $S_i \in \mathscr{S}'$, we bribe the voter with the vote of the form $p\ S_i\ \cdots$ once, so her new vote is of the form $S_i\ p\ \cdots$.

In the first round, w scores $3m$ points; p, every $x_i \in X$, and every $S_i \in \mathscr{S}'$ score $2m$ points each; every $S_i \in \mathscr{S} \setminus \mathscr{S}'$ scores $2m-1$ points; and every candidate from D and Y scores only one point. Since w is the only plurality winner, all second-place candidates (namely, p, every $x_i \in X$, and every $S_i \in \mathscr{S}'$) proceed to the second round.

In the second round, every $S_i \in \mathscr{S}'$ still scores the same number of points as in the first round, w gains $2m(m-3)$ additional points, p gains $(3m+1)+2m(m-4)$ additional points, and every $x_i \in X$ gains $(2m^2-5m-4)+4$ additional points. Therefore, p alone wins the election with $2m^2-3m+1$ points, ahead of w and every $x_i \in X$ with $2m^2-3m$ points each, and every $S_i \in \mathscr{S}'$ with $2m$ points each.

($\Leftarrow$) Suppose that $((C,V),p,B,\rho)$ is a yes-instance of Plurality-with-runoff-CONSTRUCTIVE-SHIFT-BRIBERY. Notice that if no voters are bribed, p and w are leading in the election with $3m$ points each, so they both proceed to the final round. It is easy to see that w wins against p in a one-on-one election. To prevent w and p from being the only candidates in the second round, m voters with votes of the form $p\ S_i\ \cdots$ have to be bribed. Let $\mathscr{S}' \subseteq \mathscr{S}$ be such that $S_i \in \mathscr{S}'$ if the voter with vote $p\ S_i\ \cdots$ has been bribed. Then w, p, every $x_i \in X$, and every $S_i \in \mathscr{S}'$ survive the first round. Since every other candidate is deleted in the first round, p now scores $2m^2-5m+1$ points and beats w by a margin of one point. Moreover, p beats every $S_i \in \mathscr{S}'$ since the candidates from $\mathscr{S}'$ did not gain any additional points in this round. Regarding the candidates from X, every $x_i \in X$ gains $2m^2-5m-4$ points and two additional points for every $S_j \in \mathscr{S} \setminus \mathscr{S}'$ with $x_i \in S_j$ that was eliminated in the first round. Since there are exactly three $S_j \in \mathscr{S}$ with $x_i \in S_j$, every $x_i \in X$ can gain six points if all those candidates were eliminated in the last round, which would let x_i overtake p by one point. In order for p to beat all $x_i \in X$, at least one $S_j \in \mathscr{S}$ with $x_i \in S_j$ needs to be in $\mathscr{S}'$ and is therefore still standing in the second round. Since $|\mathscr{S}'| = m$ and there are $3m$ candidates in X, p can beat every $x_i \in X$ (and subsequently win the election) only if $\mathscr{S}'$ is an exact cover of X. □

8 Conclusions and Open Questions

We have shown that shift bribery is NP-complete for each of the iterative voting systems of Hare, Coombs, Baldwin, Nanson, iterated plurality, plurality with runoff, iterated veto, and veto with runoff, each for both the constructive and the destructive case and in both the unique-winner and the nonunique-winner model. This contrasts previous results due to Elkind et al. [16, 15, 37] showing that shift bribery can be solved efficiently by exact or approximation algorithms for many natural voting rules that do not proceed iteratively. Indeed, the iterative nature of the voting rules we have studied seems to be responsible for the hardness of shift bribery.

While these are interesting theoretical results complementing earlier work both on shift bribery and on these voting systems, NP-hardness of course has its limitations in terms of providing protection against shift bribery attacks in practice. Therefore, it would be interesting to also study shift bribery for these voting systems in terms of approximation and parameterized complexity and to do a typical-case analysis.

A feature shared by most of the iterative voting rules we have studied is that many of them are not monotonic. This has the somewhat counterintuitive effect that shifting the designated candidate forward in some votes can actually hurt this candidate's chances to win, and shifting the designated candidate backward can increase these chances. We have discussed this feature in Section 7, showing that constructive shift bribery remains NP-hard even if we are allowed to only shift the designated candidate backward in some votes for two iterative voting systems: Hare voting and plurality with runoff. We leave the analogous question open for the remaining iterative voting systems studied here (except, of course, for the monotonic rules iterated plurality and iterated veto), and conjecture that they share this

property. Even more interestingly, we pose as an open question whether there is a nonmonotonic voting system—a natural one or an artificially constructed one—for which unrestricted shift bribery is NP-hard but becomes efficiently solvable when restricted to shift bribery actions specifically exploiting their nonmonotonicity (i.e., allowing to shift the designated candidate only backward in the constructive case, or forward in the destructive case).

Acknowledgements We thank the anonymous AAMAS'18 and ISAIM'18 reviewers for helpful comments.

References

1. Baldwin, J.: The technique of the Nanson preferential majority system of election. Transactions and Proceedings of the Royal Society of Victoria **39**, 42–52 (1926)
2. Bartholdi III, J., Tovey, C., Trick, M.: The computational difficulty of manipulating an election. Social Choice and Welfare **6**(3), 227–241 (1989)
3. Bartholdi III, J., Tovey, C., Trick, M.: How hard is it to control an election? Mathematical and Computer Modelling **16**(8/9), 27–40 (1992)
4. Baumeister, D., Faliszewski, P., Lang, J., Rothe, J.: Campaigns for lazy voters: Truncated ballots. In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems, pp. 577–584. IFAAMAS (2012)
5. Baumeister, D., Rothe, J.: Preference aggregation by voting. In: J. Rothe (ed.) Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division, Springer Texts in Business and Economics, chap. 4, pp. 197–325. Springer-Verlag (2015)
6. Betzler, N., Niedermeier, R., Woeginger, G.: Unweighted coalitional manipulation under the Borda rule is NP-hard. In: Proceedings of the 22nd International Joint Conference on Artificial Intelligence, pp. 55–60. AAAI Press/IJCAI (2011)
7. Brandt, F., Conitzer, V., Endriss, U., Lang, J., Procaccia, A. (eds.): Handbook of Computational Social Choice. Cambridge University Press (2016)
8. Bredereck, R., Chen, J., Faliszewski, P., Guo, J., Niedermeier, R., Woeginger, G.: Parameterized algorithmics for computational social choice: Nine research challenges. Tsinghua Science and Technology **19**(4), 358–373 (2014)
9. Bredereck, R., Chen, J., Faliszewski, P., Nichterlein, A., Niedermeier, R.: Prices matter for the parameterized complexity of shift bribery. In: Proceedings of the 28th AAAI Conference on Artificial Intelligence, pp. 1398–1404. AAAI Press (2014)
10. Bredereck, R., Faliszewski, P., Niedermeier, R., Talmon, N.: Complexity of shift bribery in committee elections. In: Proceedings of the 30th AAAI Conference on Artificial Intelligence, pp. 2452–2458. AAAI Press (2016)
11. Bredereck, R., Faliszewski, P., Niedermeier, R., Talmon, N.: Large-scale election campaigns: Combinatorial shift bribery. Journal of Artificial Intelligence Research **55**, 603–652 (2016)
12. Conitzer, V., Sandholm, T., Lang, J.: When are elections with few candidates hard to manipulate? Journal of the ACM **54**(3), Article 14 (2007)
13. Conitzer, V., Walsh, T.: Barriers to manipulation in voting. In: F. Brandt, V. Conitzer, U. Endriss, J. Lang, A. Procaccia (eds.) Handbook of Computational Social Choice, chap. 6, pp. 127–145. Cambridge University Press (2016)
14. Davies, J., Katsirelos, G., Narodytska, N., Walsh, T., Xia, L.: Complexity of and algorithms for the manipulation of Borda, Nanson's and Baldwin's voting rules. Artificial Intelligence **217**, 20–42 (2014)
15. Elkind, E., Faliszewski, P.: Approximation algorithms for campaign management. In: Proceedings of the 6th International Workshop on Internet & Network Economics, pp. 473–482. Springer-Verlag *Lecture Notes in Computer Science #6484* (2010)
16. Elkind, E., Faliszewski, P., Slinko, A.: Swap bribery. In: Proceedings of the 2nd International Symposium on Algorithmic Game Theory, pp. 299–310. Springer-Verlag *Lecture Notes in Computer Science #5814* (2009)
17. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L.: How hard is bribery in elections? Journal of Artificial Intelligence Research **35**, 485–532 (2009)
18. Faliszewski, P., Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Llull and Copeland voting computationally resist bribery and constructive control. Journal of Artificial Intelligence Research **35**, 275–341 (2009)
19. Faliszewski, P., Manurangsi, P., Sornat, K.: Approximation and hardness of shift-bribery. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, pp. 1901–1908. AAAI Press (2019)
20. Faliszewski, P., Reisch, Y., Rothe, J., Schend, L.: Complexity of manipulation, bribery, and campaign management in Bucklin and fallback voting. Journal of Autonomous Agents and Multi-Agent Systems **29**(6), 1091–1124 (2015)
21. Faliszewski, P., Rothe, J.: Control and bribery in voting. In: F. Brandt, V. Conitzer, U. Endriss, J. Lang, A. Procaccia (eds.) Handbook of Computational Social Choice, chap. 7, pp. 146–168. Cambridge University Press (2016)
22. Garey, M., Johnson, D.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman and Company (1979)
23. Gonzalez, T.: Clustering to minimize the maximum intercluster distance. Theoretical Computer Science **38**, 293–306 (1985)
24. Hemaspaandra, E., Hemaspaandra, L., Rothe, J.: Anyone but him: The complexity of precluding an alternative. Artificial Intelligence **171**(5–6), 255–285 (2007)

25. Kaczmarczyk, A., Faliszewski, P.: Algorithms for destructive shift bribery. In: Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems, pp. 305–313. IFAAMAS (2016)
26. Knop, D., Koutecký, M., Mnich, M.: Voting and bribing in single-exponential time. In: Proceedings of the 34th Annual Symposium on Theoretical Aspects of Computer Science, *LIPIcs*, vol. 66, pp. 46:1–46:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik (2017)
27. Konczak, K., Lang, J.: Voting procedures with incomplete preferences. In: Proceedings of the Multidisciplinary IJCAI-05 Workshop on Advances in Preference Handling, pp. 124–129 (2005)
28. Levin, J., Nalebuff, B.: An introduction to vote-counting schemes. The Journal of Economic Perspectives **9**(1), 3–26 (1995)
29. Maushagen, C., Neveling, M., Rothe, J., Selker, A.: Complexity of shift bribery in iterative elections. In: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, pp. 1567–1575. IFAAMAS (2018)
30. Nanson, E.: Methods of election. Transactions and Proceedings of the Royal Society of Victoria **19**, 197–240 (1882)
31. Papadimitriou, C.: Computational Complexity, second edn. Addison-Wesley (1995)
32. Porschen, S., Schmidt, T., Speckenmeyer, E., Wotzlaw, A.: XSAT and NAE-SAT of linear CNF classes. Discrete Applied Mathematics **167**, 1–14 (2014)
33. Reisch, Y., Rothe, J., Schend, L.: The margin of victory in Schulze, cup, and Copeland elections: Complexity of the regular and exact variants. In: Proceedings of the 7th European Starting AI Researcher Symposium, pp. 250–259. IOS Press (2014)
34. Rothe, J.: Complexity Theory and Cryptology. An Introduction to Cryptocomplexity. EATCS Texts in Theoretical Computer Science. Springer-Verlag (2005)
35. Rothe, J. (ed.): Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division. Springer Texts in Business and Economics. Springer-Verlag (2015)
36. Rothe, J.: Borda count in collective decision making: A summary of recent results. In: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, pp. 9830–9836. AAAI Press (2019)
37. Schlotter, I., Faliszewski, P., Elkind, E.: Campaign management under approval-driven voting rules. Algorithmica **77**, 84–115 (2017)
38. Taylor, A.: Social Choice and the Mathematics of Manipulation. Cambridge University Press (2005)
39. Xia, L.: Computing the margin of victory for various voting rules. In: Proceedings of the 13th ACM Conference on Electronic Commerce, pp. 982–999. ACM Press (2012)
40. Xia, L., Conitzer, V.: Determining possible and necessary winners given partial orders. Journal of Artificial Intelligence Research **41**, 25–67 (2011)
41. Zwicker, W.: Introduction to the theory of voting. In: F. Brandt, V. Conitzer, U. Endriss, J. Lang, A. Procaccia (eds.) Handbook of Computational Social Choice, chap. 2, pp. 23–56. Cambridge University Press (2016)

8 ZUSAMMENFASSUNG UND AUSBLICK

Wir wollen die Ergebnisse dieser Arbeit hier noch einmal kurz zusammenfassen. Darüber hinaus schauen wir uns offene Fragen an, die sich bei den untersuchten Problemen ergeben haben und diskutieren welche Richtung eine weiterführende Forschung einschlagen könnte.

In dieser Arbeit haben wir verschiedene Wahlregeln hinsichtlich diverser Kontroll- und Bestechungsprobleme untersucht. Dabei haben wir die Komplexität dieser Probleme aus der Sicht der klassischen und parametrisierten Komplexitätstheorie analysiert und in den meisten Fällen Ergebnisse in Form von Vollständigkeitsresultaten erhalten.
Teile dieser Ergebnisse wurden durch die Lektüre eines von Brandt et al. [8] verfassten Standardwerks der Computational Social Choice motiviert. In dem darin enthaltenen Kapitel zur Wahlkontrolle ist eine Übersichtstabelle dargestellt, welche für eine prominente Auswahl an Wahlregeln aufführt, wie diese sich gegenüber den von Bartholdi et al. [5] sowie Hemaspaandra et al. [38] eingeführten Kontrollproblemen verhalten. Bei Betrachtung dieser Tabelle sticht einem ins Auge, dass für die Wahlregeln Borda und Maximin noch eine Vielzahl von Ergebnissen fehlt.
Für die Wahlregel Borda haben Neveling und Rothe [61, 62] diese Lücke schließen können. Für die Wahlregel Maximin sind in der Übersichtstabelle alle Einträge bezüglich der Partitionierung der Kandidatenmenge und der Partitionierung der Wählermenge leer.
In Kapitel 6 haben wir für die Kontrollprobleme zur Partitionierung der Wählermenge die NP-Vollständigkeit für die Wahlregel Maximin nachweisen können. Bei den Kontrollproblemen zur Partitionierung der Kandidatenmenge konnten wir für die konstruktive Variante die NP-Vollständigkeit nachweisen. Zusammen mit den Ergebnissen aus der Arbeit von Maushagen und Rothe [51], wird durch die Arbeit aus Kapitel 6 die Lücke für Maximin komplett geschlossen.

Eine Wahlregel, die trotz ihrer Popularität nicht in die besagte Übersichtstabelle aufgenommen wurde, ist Veto. Nichtsdestotrotz lassen sich in der Literatur bereits einige Resultate für Veto finden, so etwa in der Arbeit von Lin [47]. Wie schon bei Maximin blieb jedoch die Frage offen, wie sich Veto gegenüber der Partitionierung der Kandidaten- und Wählermenge verhält. In Kapitel 5 haben wir uns der Beantwortung dieser Frage gewidmet. Für die Partitionierung der Kandidatenmenge haben wir gezeigt, dass alle untersuchten Varianten NP-vollständig sind. Bei den Problemen zur Partitionierung der Wählermenge ergab sich ein differenzierteres Bild. Hier hängen die Ergebnisse von der jeweils genutzten Gleichstandsbrechungsregel ab.

Wird bei einem Problem die Regel TE, also ties eliminate, genutzt, so ist dieses effizient lösbar. Bei Nutzung der Regel TP, also ties promote, ist das Problem NP-vollständig.
Nicht zuletzt wegen der Bedeutung, welche die oben erwähnte Übersichtstabelle aus dem Handbook of Computational Social Choice [8] auf die vorliegende Arbeit hatte, sei an dieser Stelle darauf hingewiesen, dass es noch vier Tabelleneinträge gibt, die auf noch offene Fragestellungen hinweisen. Zum einen ist da die Wahlregel Schulze, für welche Parkes und Xia [64], sowie Menton und Singh [57] verschiedene Kontrollprobleme hinsichtlich ihrer Komplexität klassifiziert haben. Übrig geblieben sind dabei die drei Kontrollprobleme Destructive-Control-by-Deleting-Candidates, Destructive-Control-by-Adding-an-Unlimited-Number-of-Candidates, sowie Destructive-Control-by-Adding-Candidates. Menton und Singh gehen in ihrer Arbeit davon aus, dass die Probleme wahrscheinlich effizient lösbar sind. Bisher konnte dies allerdings noch nicht bewiesen werden. Beim letzten noch offenen Tabelleneintrag handelt es sich um das Problem Destructive-Control-Partition-of-Voters-TP für die Wahlregel Bucklin. Dieses Problem ist bei der Arbeit von Erdélyi et al. [24] noch offen geblieben und stellt, wie die zuvor erwähnten Probleme, einen natürlichen Anknüpfungspunkt für die in der vorliegenden Arbeit getätigte Forschung dar.
Neben den bereits erwähnten Ergebnissen zur klassischen Komplexitätstheorie, haben wir in Kapitel 5 ein Resultat zur parametrisierten Komplexitätstheorie erhalten. Genauer haben wir das Problem Constructive-Control-by-Adding-Candidates für die Wahlregel Plurality betrachtet. Der in der Literatur angegebene Beweis zur W[1]-Härte von Chen et al. [12] erwies sich als fehlerhaft. Diesen Umstand haben wir analysiert und die W[1]-Härte mit einem neuen Beweis nachgewiesen.

In Kapitel 7 haben wir das Bestechungsproblem Shift-Bribery für insgesamt acht iterative Wahlregeln betrachtet. Dabei konnten wir für jede untersuchte Wahlregel zeigen, dass das Problem Shift-Bribery sowohl in der konstruktiven als auch in der destruktiven Variante NP-vollständig ist.
Allen Fällen gemeinsam war hierbei, dass der Nachweis der NP-Vollständigkeit bezüglich eines Typs von Preisfunktion geführt wurde. Als natürliche Frage ergibt sich, ob das Problem durch andere Preisfunktionen effizient lösbar wird.
Eine grundsätzliche Beobachtung aus der Komplexitätstheorie ist, dass sich durch den Nachweis der NP-Vollständigkeit eines Problems oft weitere, interessante Forschungsansätze eröffnen, wie etwa das Studium der parametrisierten Komplexität dieses Problems. Im Falle der in Kapitel 7 vorgestellten Arbeit [48] wurde dieser Pfad von Zhou und Guo [76] eingeschlagen. Sie haben für die Wahlregeln Hare, Coombs, Baldwin und Nanson eine parametrisierte Komplexitätsanalyse für

verschiedene Parameter durchgeführt. Sie untersuchten, welchen Einfluss das Festhalten der Parameter *Anzahl der Kandidaten*, *Anzahl der Wähler* und *Anzahl der Swap-Vertauschungen* auf die Komplexität des Problems hat. Für den Parameter Anzahl der Kandidaten hat sich dabei ergeben, dass die entsprechenden Probleme für alle betrachteten Wahlregeln sowohl in der konstruktiven als auch in der destruktiven Variante effizient lösbar sind. Wird hingegen die Anzahl der Swap-Vertauschungen fixiert, so zeigen die Autoren, dass die resultierenden Probleme W[1]-hart sind. Offen geblieben sind jedoch weiterhin einige Probleme, die sich durch das Festhalten des Parameters Anzahl der Wähler ergeben und stellen somit einen natürlichen Anknüpfungspunkt für eine weitergehende Forschung dar.
Unabhängig von den Ergebnissen der vorliegenden Arbeit, lässt sich feststellen, dass die Computational Social Choice ein sich schnell entwickelndes Forschungsgebiet ist.
Eine spannende Richtung ergibt sich bei der Betrachtung von iterativen Wahlen. Hier geht man davon aus, dass mehrere Wähler unabhängig voneinander strategisch wählen. Hier ist es häufig der Fall, dass die Wähler nur ein eingeschränktes Wissen über das Stimmungsbild haben und beispielsweise Informationen durch ein Umfrageinstitut erhalten. Wähler können dann iterativ ihre Stimme updaten und eine beste Antwortstrategie auf die momentane Situation abgeben. Hier kann man Fragen nach der Konvergenz eines solchen iterativen Prozess stellen oder beispielsweise betrachten, was passiert wenn das Umfrageinstitut versucht durch falsche Angaben die Wahl zu beeinflussen. Einen Überblick über einige Resultate zu diesem Thema kann man in dem Übersichtsartikel von Meir [55] finden.

LITERATURVERZEICHNIS

[1] K. Arrow. *Social Choice and Individual Values*. John Wiley und Sons, 1951 (revised edition 1963).

[2] K. Arrow, A. Sen und K. Suzumura, Hrsg. *Handbook of Social Choice and Welfare*. Bd. 1. North-Holland, 2002.

[3] J. Bartholdi III, C. Tovey und M. Trick. „The Computational Difficulty of Manipulating an Election“. In: *Social Choice and Welfare* 6.3 (1989), S. 227–241.

[4] J. Bartholdi III, C. Tovey und M. Trick. „Voting Schemes for Which it Can Be Difficult to Tell Who Won the Election“. In: *Social Choice and Welfare* 6.2 (1989), S. 157–165.

[5] J. Bartholdi III, C. Tovey und M. Trick. „How Hard Is It to Control an Election?“ In: *Mathematical and Computer Modelling* 16.8/9 (1992), S. 27–40.

[6] N. Boehmer, R. Bredereck, P. Faliszewski und R. Niedermeier. *On the Robustness of Winners: Counting Briberies in Elections*. Techn. Ber. arXiv:2010.09678v1. ACM Computing Research Repository (CoRR), Okt. 2020.

[7] A. Borodin, O. Lev, N. Shah und T. Strangway. „Big City vs. the Great Outdoors: Voter Distribution and How It Affects Gerrymandering.“ In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*. ijcai.org, 2018, S. 98–104.

[8] F. Brandt, V. Conitzer, U. Endriss, J. Lang und A. Procaccia, Hrsg. *Handbook of Computational Social Choice*. Cambridge University Press, 2016.

[9] R. Bredereck, J. Chen, P. Faliszewski, A. Nichterlein und R. Niedermeier. „Prices Matter for the Parameterized Complexity of Shift Bribery“. In: *Information and Computation* 251 (2016), S. 140–164.

[10] R. Bredereck, P. Faliszewski, R. Niedermeier und N. Talmon. „Large-Scale Election Campaigns: Combinatorial Shift Bribery“. In: *Journal of Artificial Intelligence Research* 55 (2016), S. 603–652.

[11] E. Brelsford, P. Faliszewski, E. Hemaspaandra, H. Schnoor und I. Schnoor. „Approximability of Manipulating Elections“. In: *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*. AAAI Press, 2008, S. 44–49.

[12] J. Chen, P. Faliszewski, R. Niedermeier und N. Talmon. „Elections with Few Voters: Candidate Control Can Be Easy“. In: *Proceedings of the 29th AAAI Conference on Artificial Intelligence*. AAAI Press, Jan. 2015, S. 2045–2051.

[13] J. Chen, I. Kanj und G. Xia. „Improved Parameterized Upper Bounds for Vertex Cover“. In: *Proceedings of the 31st International Symposium on Mathematical Foundations of Computer Science*. Springer-Verlag *Lecture Notes in Computer Science #4162*, 2006, S. 238–249.

[14] L. Chen, L. Xu, S. Xu, Z. Gao, N. Shah, Y. Lu und W. Shi. „Protecting Election from Bribery: New Approach and Computational Complexity Characterization“. In: *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Juli 2018, S. 1894–1896.

[15] A. Cohen-Zemach, Y. Lewenberg und J. Rosenschein. „Gerrymandering over Graphs“. In: *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Juli 2018, S. 274–282.

[16] S. Cook. „The Complexity of Theorem-Proving Procedures“. In: *Proceedings of the 3rd ACM Symposium on Theory of Computing*. ACM Press, 1971, S. 151–158.

[17] M. Cygan, F. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk und S. Saurabh. *Parameterized Algorithms*. Bd. 5. 4. Springer, 2015.

[18] P. Dey, N. Misra und Y. Narahari. „Frugal Bribery in Voting“. In: *Theoretical Computer Science* 676 (2017), S. 15–32.

[19] R. Downey und M. Fellows. „Fixed-Parameter Tractability and Completeness II: On Completeness for W[1]“. In: *Theoretical Computer Science* 141.1 (1995), S. 109–131.

[20] R. Downey und M. Fellows. *Parameterized Complexity*. 2nd. Springer-Verlag, 2013.

[21] E. Elkind und P. Faliszewski. „Approximation Algorithms for Campaign Management“. In: *Proceedings of the 6th International Workshop on Internet & Network Economics*. Springer-Verlag *Lecture Notes in Computer Science #6484*, Dez. 2010, S. 473–482.

[22] E. Elkind, P. Faliszewski, S. Gupta und S. Roy. „Algorithms for Swap and Shift Bribery in Structured Elections“. In: *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Mai 2020, S. 366–374.

[23] E. Elkind, P. Faliszewski und A. Slinko. „Swap Bribery“. In: *Proceedings of the 2nd International Symposium on Algorithmic Game Theory*. Springer-Verlag Lecture Notes in Computer Science #5814, Okt. 2009, S. 299–310.

[24] G. Erdélyi, M. Fellows, J. Rothe und L. Schend. „Control Complexity in Bucklin and Fallback Voting: A Theoretical Analysis“. In: *Journal of Computer and System Sciences* 81.4 (2015), S. 632–660.

[25] G. Erdélyi, E. Hemaspaandra und L. Hemaspaandra. „More Natural Models of Electoral Control by Partition“. In: *Proceedings of the 4th International Conference on Algorithmic Decision Theory*. Springer-Verlag *Lecture Notes in Artificial Intelligence #9346*, Sep. 2015, S. 396–413.

[26] G. Erdélyi, M. Nowak und J. Rothe. „Sincere-Strategy Preference-Based Approval Voting Fully Resists Constructive Control and Broadly Resists Destructive Control“. In: *Mathematical Logic Quarterly* 55.4 (2009), S. 425–443.

[27] P. Faliszewski, E. Hemaspaandra und L. Hemaspaandra. „How Hard Is Bribery in Elections?“ In: *Journal of Artificial Intelligence Research* 35 (2009), S. 485–532.

[28] P. Faliszewski, E. Hemaspaandra und L. Hemaspaandra. „Multimode Control Attacks on Elections“. In: *Journal of Artificial Intelligence Research* 40 (2011), S. 305–351.

[29] P. Faliszewski, E. Hemaspaandra, L. Hemaspaandra und J. Rothe. „Llull and Copeland Voting Computationally Resist Bribery and Constructive Control“. In: *Journal of Artificial Intelligence Research* 35 (2009), S. 275–341.

[30] P. Faliszewski, P. Manurangsi und K. Sornat. „Approximation and Hardness of Shift-Bribery“. In: *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*. AAAI Press, Juli 2019, S. 1901–1908.

[31] P. Faliszewski, Y. Reisch, J. Rothe und L. Schend. „Complexity of Manipulation, Bribery, and Campaign Management in Bucklin and Fallback Voting“. In: *Journal of Autonomous Agents and Multi-Agent Systems* 29.6 (2015), S. 1091–1124.

[32] Z. Fitzsimmons und O. Lev. „Selecting Voting Locations for Fun and Profit“. In: *Proceedings of the 29th International Joint Conference on Artificial Intelligence*. ijcai.org, Juli 2020, S. 224–230.

[33] M. Garey und D. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979.

[34] A. Gibbard. „Manipulation of Voting Schemes“. In: *Econometrica* 41.4 (1973), S. 587–601.

[35] F. Gurski, I. Rothe, J. Rothe und E. Wanke. *Exakte Algorithmen für schwere Graphenprobleme*. eXamen.Press. Springer-Verlag, 2010.

[36] S. Heinzelmann. „Der Sieg des Salamanders“. In: *Süddeutsche Zeitung* (17. Mai 2010). URL: https://www.sueddeutsche.de/politik/wahl-ohne-ueberraschung-der-sieg-des-salamanders-1.841593 (besucht am 07. 11. 2020).

[37] E. Hemaspaandra, L. Hemaspaandra und C. Menton. „Search versus Decision for Election Manipulation Problems“. In: *stacs13*. Bd. 20. LIPIcs. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Feb. 2013, S. 377–388.

[38] E. Hemaspaandra, L. Hemaspaandra und J. Rothe. „Anyone But Him: The Complexity of Precluding an Alternative“. In: *Artificial Intelligence* 171.5–6 (2007), S. 255–285.

[39] E. Hemaspaandra, L. Hemaspaandra und J. Rothe. „The Complexity of Online Manipulation of Sequential Elections“. In: *Journal of Computer and System Sciences* 80.4 (2014), S. 697–710.

[40] E. Hemaspaandra, L. Hemaspaandra und J. Rothe. „The Complexity of Online Voter Control in Sequential Elections“. In: *Journal of Autonomous Agents and Multi-Agent Systems* 31.5 (2017), S. 1055–1076.

[41] E. Hemaspaandra, L. Hemaspaandra und J. Rothe. „The Complexity of Online Bribery in Sequential Elections“. In: *Proceedings of the 17th Conference on Theoretical Aspects of Rationality and Knowledge*. Electronic Proceedings in Theoretical Computer Science #297, Juli 2019, S. 233–251.

[42] T. Ito, N. Kamiyama, Y. Kobayashi und Y. Okamoto. „Algorithms for gerrymandering over graphs“. In: *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Mai 2019, S. 1413–1421.

[43] A. Kaczmarczyk und P. Faliszewski. „Algorithms for Destructive Shift Bribery“. In: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Mai 2016, S. 305–313.

[44] R. Karp. „Reducibility among Combinatorial Problems“. In: *Complexity of Computer Computations*. Hrsg. von R. Miller und J. Thatcher. Plenum Press, 1972, S. 85–103.

[45] L. Levin. „Universal Sorting Problems“. In: *Problemy Peredaci Informacii* 9 (1973). In Russian. English translation in *Problems of Information Transmission*, 9:265–266, 1973, S. 115–116.

[46] Y. Lewenberg, O. Lev und J. Rosenschein. „Divide and Conquer: Using Geographic Manipulation to Win District-Based Elections“. In: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Mai 2017, S. 624–632.

[47] A. Lin. „Solving Hard Problems in Election Systems“. Diss. Rochester, NY, USA: Rochester Institute of Technology, März 2012.

[48] C. Maushagen, M. Neveling, J. Rothe und A. Selker. „Complexity of Shift Bribery in Iterative Elections“. Manuscript submitted to *Journal of Autonomous Agents and Multi-Agent Systems*.

[49] C. Maushagen, M. Neveling, J. Rothe und A. Selker. „Complexity of Shift Bribery in Iterative Elections“. In: *Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems*. A preliminary version was presented at ISAIM 2018. IFAAMAS, Juli 2018, S. 1567–1575.

[50] C. Maushagen und J. Rothe. „Complexity of Control by Partitioning Veto and Maximin Elections and of Control by Adding Candidates to Plurality Elections“. In: *Proceedings of the 22nd European Conference on Artificial Intelligence*. A preliminary version was presented at ISAIM 2016. IOS Press, Aug. 2016, S. 277–285.

[51] C. Maushagen und J. Rothe. „Complexity of Control by Partitioning Veto and Maximin Elections“. In: *Nonarchival website proceedings of the 14th International Symposium on Artificial Intelligence and Mathematics*. Jan. 2016. URL: http://isaim2016.cs.virginia.edu/papers/ISAIM2016_Maushagen_Rothe.pdf.

[52] C. Maushagen und J. Rothe. „Complexity of Control by Partition of Voters and of Voter Groups in Veto and Other Scoring Protocols“. In: *Proceedings of the 16th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, 2017, S. 615–623.

[53] C. Maushagen und J. Rothe. „Complexity of Control by Partitioning Veto Elections and of Control by Adding Candidates to Plurality Elections“. In: *Annals of Mathematics and Artificial Intelligence* 82.4 (2018), S. 219–244.

[54] C. Maushagen und J. Rothe. „The Last Voting Rule Is Home: Complexity of Control by Partition of Candidates or Voters in Maximin Elections“. In: *Proceedings of the 24th European Conference on Artificial Intelligence*. IOS Press, 2020, S. 163–170.

[55] R. Meir. „Iterative voting“. In: *Trends in Computational Social Choice*. Hrsg. von U. Endriss. AI Access Foundation, 2017. Kap. 4, S. 69–86.

[56] C. Menton. „Normalized Range Voting Broadly Resists Control“. In: *Theory of Computing Systems* 53.4 (2013), S. 507–531.

[57] C. Menton und P. Singh. „Control Complexity of Schulze Voting“. In: *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*. AAAI Press/IJCAI, Aug. 2013, S. 286–292.

[58] P. Middelhoff. „Ist die US-Wahl gezinkt?“ In: *Zeit Online* (28. Okt. 2016). URL: https://www.zeit.de/politik/ausland/2016-10/betrugsvorwurf-usa-wahlen-praesidentschaft-donald-trump/komplettansicht (besucht am 07. 11. 2020).

[59] S. Misteli. „Obama geht auf Salamanderjagd“. In: *Neue Zürcher Zeitung* (2. Okt. 2017). URL: https://www.nzz.ch/international/barack-obama-macht-wieder-politik-ld.1319540 (besucht am 07. 11. 2020).

[60] E. Müller. „Ich zieh mir die Grenze, wie sie mir gefällt ...“ In: *Frankfurter Rundschau* (23. Okt. 2020). URL: https://www.fr.de/politik/gerrymandering-in-den-usa-ich-zieh-mir-die-grenze-wie-sie-mir-gefaellt-90079071.html (besucht am 07. 11. 2020).

[61] M. Neveling und J. Rothe. „Closing the Gap of Control Complexity in Borda Elections: Solving Ten Open Cases“. In: *Proceedings of the 18th Italian Conference on Theoretical Computer Science*. Bd. 1949. CEUR-WS.org, Sep. 2017, S. 138–149.

[62] M. Neveling und J. Rothe. „Solving Seven Open Problems of Offline and Online Control in Borda Elections“. In: *Proceedings of the 31st AAAI Conference on Artificial Intelligence*. AAAI Press, Feb. 2017, S. 3029–3035.

[63] C. Papadimitriou. *Computational Complexity*. Second. Addison-Wesley, 1995.

[64] D. Parkes und L. Xia. „A Complexity-of-Strategic-Behavior Comparison between Schulze's Rule and Ranked Pairs“. In: *Proceedings of the 26th AAAI Conference on Artificial Intelligence*. AAAI Press, Juli 2012, S. 1429–1435.

[65] Y. Reisch, J. Rothe und L. Schend. „The Margin of Victory in Schulze, Cup, and Copeland Elections: Complexity of the Regular and Exact Variants“. In: *Proceedings of the 7th European Starting AI Researcher Symposium*. IOS Press, Aug. 2014, S. 250–259.

[66] H. Riemenschneider. „Münster künftig in drei Wahlkreise geteilt“. In: *Westfäliche Nachrichten* (7. Nov. 2020). URL: https://www.wn.de/Muenster/4308866-Gesetzentwurf-Muenster-kuenftig-in-drei-Wahlkreise-geteilt (besucht am 07. 01. 2021).

[67] J. Rothe. *Komplexitätstheorie und Kryptologie: Eine Einführung in Kryptokomplexität*. Springer-Verlag, 2008.

[68] J. Rothe, Hrsg. *Economics and Computation. An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*. Springer Texts in Business and Economics. Springer-Verlag, 2015.

[69] M. 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), S. 187–217.

[70] I. Schlotter, P. Faliszewski und E. Elkind. „Campaign Management under Approval-Driven Voting Rules“. In: *Proceedings of the 25th AAAI Conference on Artificial Intelligence*. AAAI Press, Aug. 2011, S. 726–731.

[71] F. Steffens. „Eine Demokratie mit Macken“. In: *Frankfurter Allgemeine* (27. Jan. 2018). URL: https://www.faz.net/aktuell/politik/von-trump-zu-biden/usa-streit-ueber-gerrymandering-und-das-wahlsystem-15414307.html (besucht am 07. 11. 2020).

[72] A. Taylor. *Social Choice and the Mathematics of Manipulation*. Cambridge University Press, 2005.

[73] C. Thiele. *Regeln und Verfahren der Entscheidungsfindung innerhalb von Staaten und Staatenverbindungen: Staats-und kommunalrechtliche sowie europa-und völkerrechtliche Untersuchungen*. Springer Science & Business Media, 2008.

[74] L. Xia. „Computing the Margin of Victory for Various Voting Rules“. In: *Proceedings of the 13th ACM Conference on Electronic Commerce*. ACM Press, Juni 2012, S. 982–999.

[75] Y. Yang, Y. Shrestha und J. Guo. „On the Complexity of Bribery with Distance Restrictions“. In: *Theoretical Computer Science* 760 (2019), S. 55–71.

[76] A. Zhou und J. Guo. „Parameterized Complexity of Shift Bribery in Iterative Elections“. In: *Proceedings of the 19th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, Mai 2020, S. 1665–1673.

EIDESSTATTLICHE ERKLÄRUNG

ENTSPRECHEND §5 DER PROMOTIONSORDNUNG VOM 15.06.2018

Ich versichere an Eides Statt, dass die Dissertation von mir selbständig und ohne unzulässige fremde Hilfe unter Beachtung der „Grundsätze zur Sicherung guter wissenschaftlicher Praxis an der Heinrich-Heine-Universität Düsseldorf“ erstellt worden ist.
Des Weiteren erkläre ich, dass ich eine Dissertation in der vorliegenden oder in ähnlicher Form noch bei keiner anderen Institution eingereicht habe.

Teile dieser Arbeit wurden bereits in den folgenden Schriften veröffentlicht oder zur Begutachtung eingereicht: [50], [52], [53], [54], [49] und [48].

Düsseldorf, Februar 2021

Cynthia Maushagen