# Manipulation, Control, Bribery and Barriers

**Why rigging an election is too hard?**

Arjun Maneesh Agarwal, Chennai Mathematical Institute

# Outline

# Computational Social Science

# Introduction

- Social Science is the study of societies and the relationships among members within those societies. We often have anthropology, archaeology, economics, geography, history, linguistics, management, communication studies, psychology, culturology, and **political science.**
- Decision theory or Choice Theory, at the heart of it, is the math of modeling real-life scenarios and optimizing decision-making. It deals with financial markets and elections, fairness and moral hazard, auctions and corruption; everywhere a decision is to be made, decision theory enters.
- Computer Science is, at the heart of it, concerned with which problems can we solve and how fast can we solve them. Some branches also deal with the question of how well can we approximate the answer or given some solution, solve another problem. These problems are generally mathematical in nature.

Thus,

# Introduction (ii)

$$\text{Social Science} \xrightarrow{\text{Decision Theory}} \text{Computer Science}$$

The image under this function is referred to as "Computational Social Science" or "Computational Social Choice" aka COMSOC.

# Voting Theory

- Treating voting rules as an algorithm
- While this may seem like an oblivious idea, it was quite a revolutionary idea 30 years back.
- Furthermore, this acts like a five year old idea. We will see a lot of rather common things which are open.

# Let’s Begin

# Voting Theory (iii)

**Definition 1:** The Vote Aggregation Problem.

Problem: Given $n$ alternatives, and $v$ voters, each with an preference ordering on the alternatives, aggregate them into either

1. a "winner" or winners, or
2. a total ordering of the alternatives (where the "winner" is the first in the ordering).

Clearly a huge number of voting rules satisfies this definition.

However, most of them are not decisive, fair and 'practical'.

# Anonymity, Neutrality and Monotonicity

**Definition 2.**

- A voting rule is **anonymous** if it treats all of the voters equally, meaning that if any two voters traded votes, the outcome of the election would remain the same.
- A voting rule is **neutral** if it treats all the candidates equally, meaning that if every voter switched them on their votes, the outcome of the election would change accordingly.
- A voting rule is **monotone** if it is impossible for a candidate to become rated worse in the societal ordering by gaining votes (or moving above in the rankings of the voters).

A weaker version of Monotonicity is called Unanimity.

# Anonymity, Neutrality and Monotonicity (ii)

**Corollary 1.**

A voting system is **unanimous** if whenever every voter prefers candidate $c$ to $d$, then $d$ is not the winner (and is ranked below $c$ in the societal ordering).

# Fairness

I will now propose a conditions which I believe everyone will agree are desirable

**Definition 3.**

- **Independence of Irrelevant Alternatives**: A voting system satisfies independence of irrelevant alternatives if the decision of $c$ versus $d$ depends only on the relative ranking of $c$ and $d$ in the voter preference profiles.

Congratulations! We have just walked into an impossibility.

**Theorem 1:** Arrow's Theorem.

The only voting rule that satisfies Unanimity and Independence of Irrelevant Alternatives is Dictatorship

A proof for this was presented by Krutarth Shah in his seminar. I will not be sharing a proof for it today, but it is a relativly nice proof and hopefully everyone can google it in their time.

# Practicality

- How quickly can we determine the result under a particular voting rule? $n$ candidates, $v$ voters.
- For Plurality: It is $\mathcal{O}(n)$
- For Borda (and other scoring rules): It is $\mathcal{O}(nv)$
- For STV: It is $\mathcal{O}(n^2)$

Note, even larger polynomial algorithms are also not really practical. In a nation as big as India, an election with complexity say $\mathcal{O}(v^4)$ would be too slot to run.

# Impracticality Theorem

An axiom which seems rather innocuous is

**Definition 4:** Consistency.

Consistency: A voting system is consistent if, when two disjoint sets of voters agree on a candidate $c$, the union of voters will also choose $c$.

Formally, if $A, B$ are the set of voters and $f$ is a voting rule satisfying consistency then $A \cap B = \emptyset$, $f(A) \cap f(B) \neq \emptyset \Rightarrow f(A \cup B) = f(A) \cup f(B)$.

but this leads to:

**Theorem 2:** BTT (Bartholdi, Tovey, and Trick) Impracticality Theorem.

For any voting system that satisfies
(a) neutrality
(b) consistency
(c) Condorcet winner
it is NP-Hard to determine the winner.

# Impracticality Theorem (ii)

***Proof.***

**Definition 5:** Kemeny's Rule.

Kemeny Rule selects the ordering that is "closest" to the voters' preferences.

Formally,

$$f(\{p_1, p_2, ..., p_v\}) = \arg \min_{\sigma \in \operatorname{perm}(n)} \sum_{i \in [v]} \mathrm{KT}(\sigma, p_i)$$

where $\mathrm{KT}(\sigma, \tau) = |\{(a, b) \mid a, b \in [n], a > b \in \sigma, b > a \in \tau\}|$.

As multiple permutations can be the Kemeny orderings, we don't really tiebreak and let the output be a set.

# Impracticality Theorem (iii)

**Lemma 1:** (HP Young and Levenglick).

The only voting system that satisfies
(a) neutrality
(b) consistency
(c) Condorcet winner
is Kemeny Rule.

**Lemma 2:** (Bartholdi, Tovey, and Trick).

Kemeny score is NP-complete, and Kemeny ranking and Kemeny winner are NP-hard.

It is easy to see how the these lemmas lead to the claimed theorem. The proof for both is in the appendix.

# Manipulation

# Example

- Consider an election with three alternatives, $a$, $b$ and $c$,and three voters, 1, 2 and 3.
- Suppose the rule used is plurality with ties broken toward alternatives earlier in the alphabet.
- Suppose voter 3 knows (or strongly suspects) that voter 1 will rank $a$ first in her vote, and that voter 2 will rank $b$ first. Voter 3's true preferences are $c \succ b \succ a$.
- If she votes truthfully, this will result in a three-way tie, broken in favor of $a$ which is 3's least preferred alternative. If, instead, voter 3 ranks $b$ first, then $b$ will win instead. Hence, voter 3 has an incentive to cast a vote that does not reflect her true preferences.

Here 3 has misrepresented her true preferences to affect the outcome of the election. This is called a 'manipulation'.

# Two Negative Results

**Theorem 3:** Gibberd-Satterwaite.

Consider a (resolute) voting rule that is defined for some number $m$ of alternatives with $m \geq 3$, with no restrictions on the preference domain. Then, this rule must be at least one of the following:

- Dictatorial: there exists a single fixed voter whose most-preferred alternative is chosen for every profile;
- Imposing: there is at least one alternative that does not win under any profile;
- manipulable (i.e., not strategyproof).

The proof is quite similar to Arrow's case bash proof. We will omit it for brevity.

# A ray of hope

As we saw, even finding the winner to an fair election can be hard. That means, once could reasonably expect there to be election rules where finding the manipulation is NP hard.

# Manipulation Problem

Given A profile $\Pi$ of votes cast by everyone else and the preferred alternative $a$, can we vote in a way so that $a$ wins?

**Definition 6:** BTT Voting Rules.

We say a voting rule is BTT if

1. it can be run in polynomial time
2. for every $\Pi$ and every alternative it assigns a score $S(\Pi, a)$ to $a$.
3. For every profile $\Pi$, the maximum scoring alternative wins.
4. For any $\Pi$ and $\Pi'$ and alternative $a$, if $\left\{b : a \underset{i}{\succ} b\right\} \subseteq \left\{b : a \underset{i}{\succ}' b\right\}$ for every voter $i$, then $S(\Pi, a) \leq S(\Pi', a)$.

**Theorem 4:** Manipulation of BTT .

The Manipulation problem can be solved in polynomial time for any BTT rule.

The answer is literally the greedy algorithm.

# Manipulation Problem (ii)

```
Function FindManipulativeVote(alternatives, profile, a,
scoring_function):
    vote = []
    vote.append(a)

    remaining_alternatives = alternatives.remove(a)

    while remaining_alternatives is not empty:
        manipulation_found = False

        for b in remaining_alternatives:
            temp_vote = vote.copy()
            temp_vote.append(b)
            temp_vote += (remaining_alternatives - {b})

            S_a = scoring_function(a, profile + [temp_vote])
            S_b = scoring_function(b, profile + [temp_vote])
```

# Manipulation Problem (iii)

```
            if S_a > S_b:
                vote.append(b)
                remaining_alternatives.remove(b)
                manipulation_found = True
                break   // Proceed to next rank position

        if not manipulation_found:
            Report("Manipulation is not possible.")
            return None
    return vote
```

What about some natural rules that don’t come under the BTT?

# Manipulation Problem (iv)

**Definition 7:** Copeland.

The Copeland score of a candidate is the number of pairwise contests won minus the number lost.

The Second order Copeland score of a candidate is the sum of the Copeland scores of each defeated candidate.

We define n-Copeland score in a similar fashion. We define n-Copeland rule as choosing the candidate with highest n-Copeland score.

While first order Copeland clearly is BTT, second order is not. And in Appendix B, you can see a reduction from SAT to second Copeland.

Similarly, a big favorite in the community,

# Manipulation Problem (v)

**Definition 8:** Single Transferable Vote.

At each stage, the alternative with lowest plurality score is dropped from all ballots, and at the first stage for which some alternative $x$ sits atop a majority of the ballots, $x$ is declared the winner.

is also NP hard to manipulate by a very similar proof to Copeland.

# Coalition of Manipulators

Here is the thing, running a random analysis will give one that in most cases, a single voter can't really manipulate the election. This is increasingly unlikely as the number of voters increase. Just as no con happens alone, what if the election is to be stolen by a group?

Well, the cooperation makes it much harder here.

**Theorem 5.**

Unweighted coalition manipulation for the Borda rule is NP-complete with two manipulators.

# Coalition manipulation of Borda in in NP

**Proof** We will show a reduction from Permutation Sum problem.

**Definition 9:** Permutation Sum.

Given $n$ integers $X_1 \leq X_2 \leq \ldots \leq X_n$ where $\sum_{i=1}^{n} X_i = n(n+1)$, does there exist two permutations $\sigma$ and $\pi$ of $1 \ldots n$ such that $\sigma(i) + \pi(i) = X_i$ for $1 \leq i \leq n$.

Let the candidates be in favor of candidate 1 and say the tie break favors candidate 1.

**Lemma 3.**

Given integers $X_1$ to $X_m$, there exist votes over $m + 1$ candidates and a constant $C$ such that the final score of candidate $i$ is $X_i + C$ for $1 \leq i \leq m$ and for candidate $m + 1$ is $y \leq C$.

**Proof** We show how to increase the score of a candidate by 1 more than the other candidates except for the last candidate whose score increases by 1 less. For instance, suppose we wish to increase the score of candidate 1 by 1 more

# Coalition manipulation of Borda in in NP (ii)

than candidates 2 to $m$ and by 2 more than candidate $m + 1$. Consider the following pair of votes:

$$1 > m + 1 > 2 > \ldots > m - 1 > m$$
$$m > m - 1 > \ldots > 2 > 1 > m + 1$$

The score of candidate 1 increases by $m + 1$, of candidates 2 to m by $m$, and of candidate m + 1 by $m - 1$. By repeated use of this, we can achieve the result we desire. ■

Using the Lemma, we can construct the non-manipulators such that the score vector is

$$\langle C, 2(n + 2) - X_1 + C, \ldots, 2(n + 2) - X_n + C, 2(n + 2) + C, y\rangle$$

We claim two manipulators can make candidate 1 win if and only if the permutation sum problem has a solution.

($\Rightarrow$) As a permutation sum exists, the manipulators can vote as:

# Coalition manipulation of Borda in in NP (iii)

$$\langle n+2, \sigma(1), ..., \sigma(n), 0, n+1\rangle$$
$$\langle n+2, \pi(1), ..., \pi(n), 0, n+1\rangle$$

getting the score to:

$$\langle 2(n+2)+C, 2(n+2)+C, ..., 2(n+2)+C, 2(n+1)+y\rangle$$

and thus, getting candidate 1 to win.

($\Leftarrow$) To ensure candidate 1 beats candidate $n+2$, both manipulators must put candidate 1 in first place and the latter in last.

Candidate 1 in future is above the $i+1$-th candidate by $X_i$ votes where $\sum_{i=1}^{n} X_i = n(n+1)$. This means that if any of them get the score addition of $n+1$, candidate 1 will lose. So, $n+1$ scores will have to go to the last (and least dangerous) candidate.

This makes the manipulated votes of the form

# Coalition manipulation of Borda in in NP (iv)

$$\langle n+2, \sigma(1), ..., \sigma(n), 0, n+1\rangle$$
$$\langle n+2, \pi(1), ..., \pi(n), 0, n+1\rangle$$

where $\sigma$ and $\pi$ are permutations of $1...n$. To ensure candidate 1 beats candidate $i+1$, we must have $\sigma(i)+\pi(i) \leq X_i$. Since $\sum_{i=1}^{n} \sigma(i) = \frac{n(n+1)}{2}$ and $\sum_{i=1}^{n} \pi(i) = \frac{n(n+1)}{2}$; we must have $\sigma(i)+\pi(i) = X_i$.

This means, we have a solution of the permutation sum problem. ■

Similar proofs hold for Copeland etc.

# Control

# Definitions

As manipulation is the voters trying to, well, manipulate the election; Control refers to the people in power, called chair, trying to, well, control the election by adding or removing voters or candidates.

**Definition 10:** Constructive Control by Adding an Unlimited Number of Candidates.

Let $f$ be a voting rule. In the Constructive Control by Adding an Unlimited Number of Candidates problem for $f$ ($f$-CCAUC), we are given:

1. A set $A$ of qualified candidates, a set $B$ of spoiler candidates, where $A \cap B = \varphi$, and an election $(A \cup B, R)$.
2. A preferred candidate $p \in A$.

We ask if we can choose a subset $B' \subseteq B$ of the spoiler candidates such that $p$ is the unique $f$-winner of the election $(A \cup B', R)$.

# Definitions (ii)

**Definition 11:** Constructive Control by Adding Candidates.

Constructive Control by Adding Candidates problem for $f$ ($f$-CCAC) is defined similarly: In addition to (1) and (2) we are also given (3) a bound $k \in \mathbb{N}$, and we ask if there is a subset $B' \subseteq B$ of spoiler candidates such that $|B'| \leq k$ and $p$ is the unique $f$-winner of $(A \cup B', R)$.

**Definition 12:** Constructive Control by Deleting Candidates.

In the Constructive Control by Deleting Candidates problem for $f$ ($f$-CCDC), we are given :

1. An election $(A, R)$.
2. A preferred candidate $p \in A$.
3. A bound $k \in \mathbb{N}$.

We ask if $p$ can be made a unique $f$-winner of the election resulting from $(A, R)$ by deleting at most $k$ candidates.

# Definitions (iii)

**Definition 13:** Constructive Control by Adding Voters.

Let $f$ be a voting rule. In the Constructive Control by Adding Voters problem for $f$ ($f$-CCAV), we are given:

1. a list $R$ of already registered votes, a list $S$ of as yet unregistered votes, and an election $(A, R + S)$, where "profile addition" means concatenation of profiles.
2. A preferred candidate $p \in A$.
3. A bound $k \in \mathbb{N}$.

We ask if we can choose a sublist $S' \subseteq S$ of size at most $k$ such that $p$ is the unique $f$-winner of $(A, R + S')$.

# Definitions (iv)

**Definition 14:** Constructive Control by Deleting Voters.

In the Constructive Control by Deleting Voters problem for $f$ ($f$-CCDV), we are given:

1. An election $(A, R)$.
2. A preferred candidate $p \in A$
3. A bound $k \in \mathbb{N}$.

We ask if we can make $p$ a unique $f$-winner of the election resulting from $(A, R)$ by deleting no more than $k$ votes.

We are interested in these forms of control as they are quite common. From the Bush elections where Nader was a spoiler candidate to Miller (2008) showing that enfranchisement of woman voters (in many nations) was beneficial for candidates who wanted to increase spending on Infant Health.

We also see this in the papal conclave where Pope Paul VI brought the rule that cardinals above 80 may not participate in the conclave; effectively disenfranchising a lot of the conservative cardinals appointed by Pope John

# Definitions (v)

XXIII and Pius XII. We also saw this in the 2025 papal conclave where Pope Francis controlled the election from beyond the veil as he appointed 110 cardinals (of the 135 candidates and voters).

**Definition 15:** Immunity and Susceptibility.

Depending on the voting rule, it may never (for no preference profile at all) be possible for the chair to successfully exert some control action (e.g., constructive control by deleting voters) in the sense that $p$ can be turned (by deleting voters) from not being a unique winner into being one. If that is the case, we say this voting rule is **immune** to this type of control.

Otherwise (i.e., if there is at least one preference profile where the chair can successfully exert this control action), we say this voting rule is **susceptible** to this type of control.

# Definitions (vi)

**Definition 16:** Vulnerability and Resistance.

$f$ is said to be **vulnerable** (respectively, **resistant**) to this control type if the corresponding problem (e.g., f-CCAV) is in P (respectively, NP-hard).

# Analysis

Immunity appears to be very desirable.

However, Immunity for candidate control is rare. This is due to the study of candidate-voter models.

We can consider a setting where the candidates have preferences regarding election outcomes, and can strategically choose to join the race or not. For most typical election rules there are settings where some candidates would prefer not to participate in the election. In effect, such rules cannot be immune to candidate control. Nonetheless, in some rare cases (Condorcet and approval voting) immunity results for candidate control hold.

For the case of voter control, immunity is not only rare, but also is utterly undesirable. Indeed, it is natural to expect that if we add sufficiently many voters with the same preference order, then their most preferred candidate becomes a winner.

# Some Results

**Theorem 6.**

1. Condorcet and approval voting are immune and plurality is resistant to constructive control by adding (respectively, adding an unlimited number of) candidates.
2. Condorcet and approval voting are vulnerable and plurality is resistant to constructive control by deleting candidates.
3. Condorcet and approval voting are resistant and plurality is vulnerable to constructive control by both adding and deleting voters.

These results follow from the fact Condorcet and approval voting satisfy the Weak Axiom of Revealed Preference, which states that a unique winner $p$ in a set $A$ of alternatives always is also a unique winner among each subset containing $p$.

The vulnerability claims follow from a simple greedy algorithm.

The resistance here follows from reduction from Exact Cover by 3 Sets.

# Some Results (ii)

**Adding Voters**: We will show reduction from exact 3 cover or X3C. Let $B = \{b_1, b_2, ..., b_m\}, m = 3k$ and $S = \{S_1, S_2, ..., S_n\}$ such that $S_i \subset B, |S_i| = 3$.

Take the candidates of an election to be $B \cup \{w\}$. We want $w$ to win. We have $k - 2$ franchised voters who approve of $b_1, ..., b_m$ and disapprove of $w$.

We have $n$ unregistered voters where for $1 \leq i \leq n$, $v_i$ approves of $S_i \cup \{w\}$ and disapproves of everyone else.

We can enfranchish $k$ voters with ties broken against $\{w\}$ (that is in case of tie, $w$ loses).

($\Longrightarrow$) Simply enfranchising the $k$ voters that correspond to the exact cover for $B$ gets $w$'s score to $k$ and every $b \in B$ has $(k - 2) + 1 = k - 1$ score votes, so $w$ is the winner.

# Some Results (iii)

($\Longleftarrow$) Any addition which gets $w$ to win can only give 1 vote to another candidate and we need to add $k$ voters. Thus, we will have to provide an exact cover.

**Deleting Voters**: We will show reduction again from X3C. Let $B = \{b_1, b_2, ..., b_m\}, m = 3k$ and $S = \{S_1, S_2, ..., S_n\}$ such that $S_i \subset B, |S_i| = 3$. For all $1 \leq j \leq m$, let $l_j$ denote the number of $S$, $b_j$ is an element of.

Take the candidates of the election of $B \cup \{w\}$. We want $w$ to win.

We have $n$ voters such that for $1 \leq i \leq n$ such that $v_i$ approves of candidates in $S_i$ and nobody else. We also have $n$ more voters such that for $1 \leq i \leq n$, $v_i$ approves of $w$ and $b_j$ if $i \leq n - l_j$.

We can disfrancish $k$ voters with ties broken against $\{w\}$ (that is in case of tie, $w$ loses).

Notice, in the initial election, all candidates have score $n$.

# Some Results (iv)

($\Longrightarrow$) If we have an exact cover, then we can disenfrancise the voters corresponding to the cover, and thus, $w$ will have score $n$ and all other candidates have $n - 1$ score.

($\Longleftarrow$) Suppose that $w$ can be made the approval winner by deleting at most $k$ voters.

WLOG, we may assume that none of the deleted voters approves of $w$ (as otherwise, adding them will also have $w$ win). So, we assume that only voters corresponding to $S_i$'s have been deleted. For $w$ to have become winner, every $b \in B$ must have lost at least one vote. It follows that the deleted voters correspond to a cover, and since the cover has size at most $k$, this must be an exact cover for $B$.

# Bribery

# Bribery

Bribery is almost opposite to control as in this case it is not possible to affect the structure of the election at hand (that is, the sets of candidates or voters cannot be changed), but it is possible to change some of the votes instead.

The briber's task has two main components. First, the briber needs to decide who to bribe. Second, the briber has to decide how to change the chosen votes. So in some way bribery combines a control-like action (picking which voters to affect) with a manipulation-like action (deciding how to change the selected votes).

# Definitions

We begin with a complete definition which we will add some more constraints on them to make them more fit for algorithmic analysis.

**Definition 17:** Bribery Problem.

Let $f$ be a voting rule.

**Input:** An election $(A, R)$ with $N = \{1, ..., n\}$ voters with preference orders $\underset{i}{\prec}$ making up $R$, a preferred alternative $p \in A$, a budget $B \in \mathbb{N}$ and a collection of price functions $\Pi = \{\pi_1, ..., \pi_n\}$ where for each $i$, $1 \leq i \leq n$, for each preference order $\prec$, $\pi_i(\prec)$ is the cost of convincing voter $i$ to cast vote $\prec$. Note, $\pi_i\left(\underset{i}{\prec}\right) = 0$

We ask if there is a preference profile $R' = \left(\underset{1}{\prec}', ..., \underset{n}{\prec}'\right)$ such that $f(A, R') = p$ and $\sum_{i=1}^{n} \pi_i\left(\underset{i}{\prec}'\right) \leq B$.

The problem is that defining these cost functions will require quite a lot of information.

# Definitions (ii)

Hence, there are 3 stipulations we consider:

**Definition 18:** Discrete Bribery.

$$\forall 1 \leq i \leq n,$$

$$\pi_i(\prec) = \begin{cases} 0 \text{ if } \prec = \underset{i}{\prec} \\ 1 \text{ otherwise} \end{cases}$$

**Definition 19:** Dollar Discrete Bribery.

$$\pi_i(\prec) = \begin{cases} 0 \text{ if } \prec = \underset{i}{\prec} \\ c_i \text{ otherwise} \end{cases}$$

Note: each voter can have a different $c$.

# Definitions (iii)

**Definition 20:** Swap Bribery.

$1 \leq i \leq n$ and $x, y \in A$, we define $c_i^{x,y}$ as the cost to switch these candidates for voter $i$. Thus, $\pi_i(\prec)$ is the sum of all costs for $x, y$ ranked opposite $\underset{i}{\prec}$.

# Results on Plurality

**Theorem 7.**

For plurality,

1. Bribery, Weighted Bribery and Dollar Bribery are each in P.
2. Weighted Dollar Bribery is NP-complete.
3. Swap Bribery is in P.

A simple greedy algorithm solves bribery and dollar bribery. Till budget lasts or $p$ wins the election, change the vote of the cheapest to manipulate voter of one of the winning candidates to $p$.

Unfortunately, such greedy approaches do not work for Weighted-Bribery. For example, consider an algorithm that works in iterations and in each iteration bribes the heaviest voter among those that vote for one of the current winners.

# Results on Plurality (ii)

Let $(A, R)$ be an election where $A = \{p, a, b, c\}$ and where we have nine weight 1 voters voting for $a$, a single weight 5 voter voting for $b$, and a single weight-5 voter voting for $c$.

Clearly, it suffices to bribe the two weight-5 voters, but the heuristic would bribe five voters with weight 1 each.

On the other hand, bribing the heaviest voter first does not always work either. Say $A = \{p, a, b\}$, $p$ receiving no votes at first, $a$ receiving three weight-2 votes and one weight-1 vote, and $b$ receiving two weight-3 votes; to make $p$ a winner it suffices to bribe one weight-2 vote and one weight-3 vote, but the heuristic bribes three votes.

Nonetheless, a combination of these two heuristics does yield a polynomial-time algorithm.

# Results on Plurality (iii)

The idea is to find an algorithm in P time using some parameter and then find the parameter in P time as well. Here the parameter is the least amount of points $p$ must end up with after the bribery is done. Let's call it $T$.

Naturally, all the other alternatives have to end up with at most $T$ points. Thus, for each alternative a that has more than $T$ points, we should keep bribing its heaviest voters until its score decreases to at most $T$ *(this corresponds to running the bribe the current winner's heaviest voter heuristic).*

If, after bringing each alternative to at most $T$ points, the preferred alternative still does not have $T$ points, we bribe the globally heaviest voters to vote for the preferred alternative. We do so until the preferred alternative reaches at least T points *(this corresponds to running the bribe the heaviest voter heuristic).*

For each alternative $a$, we bribe $a$'s voters in the order of their nonincreasing weights. Thus, after executing the above-described strategy for some optimal

# Results on Plurality (iv)

value $T$ , $a$'s score is in the set

$\{a\text{'s original score}, a\text{'s score without its heaviest voter}, a\text{'s score without its tw}$

Thus it suffices to consider values $T$ of this form only (for each candidate)

and to pick one that leads to a cheapest bribery.

# Results on Plurality (v)

Swap-Bribery requires a somewhat different approach. The reason is that under Swap-Bribery it might not always be optimal to push our preferred candidate to the top of the votes, but sometimes it may be cheaper and more effective to replace some high-scoring alternatives with other, low-scoring ones. To account for such strategies, Elkind et al. (2009c) compute, for each vote $v$, the lowest cost of replacing $v$'s current top-alternative with each other one, and then run a flow-based algorithm.[1]

[1] What does a flow based algorithm mean? Is this related to Max Cut-Min Flow?

# Results on Plurality (vi)

Dollar Weighted-Bribery is easily shown to be NP-complete thanks to the partition problem. Given a sequence $s_1, s_2, \ldots, s_n$; we can make a 2 candidate election with all the voters voting for the candidate we don't prefer. Weight of vote and cost of bribery is both $s_i$ for the $i$th voter. We have a budget of $\frac{s_1+s_2+\ldots+s_n}{2}$.

# Conclusion

# Why does all of this matter?

A question one might ask is why prevent manipulation in the first place?

Consider a plurality election with three alternatives. If one of the candidates is considered to have a poor chance of winning the election (consider, for example, a third party in the United States), then everyone might vote for one of the other two candidates, in order to avoid wasting their votes. Is this a significant problem? Will it not simply result in the same winner that plurality-with-runoff (or STV) would have chosen (if everyone had voted truthfully), and is that so bad?

This is a topic which has revived debate in both the COMSOC and Political Science communities. Some arguments for preventing manipulation are:

- Bad equilibria. It is not at all clear that the resulting winner will be the same as the true plurality-with-runoff winner. All that is required is that voters expect the third party to have poor chances. It is possible that this alternative is actually very much liked across the electorate, but nobody is aware of this. Even more strikingly, it is possible that everyone is aware of

# Why does all of this matter? (ii)

this, and yet the alternative is expected to perform poorly—for example, because nobody is aware that others are aware of the alternative's popularity. Hence, an alternative that is very much liked, and perhaps would have won under just about any reasonable rule had everyone voted truthfully, may not win.

- Lack of information. Even if the bad equilibria described above are in fact avoided, we cannot be sure that this is the case, because we will never know exactly how popular that third alternative really was. This also interferes with the process of identifying more desirable alternatives in the next election.
- Disenfranchisement of unsophisticated voters. Voters who are less well informed may end up casting less effective votes than those who are well informed (for example, votes for the third alternative). Knowledge is power —but in many elections, this is not considered desirable.
- Wasted effort. Even if all agents manipulate to the same extent, still much effort, whether of the computational, information gathering, or

# Why does all of this matter? (iii)

> communicational variety, is expended in figuring out how to manipulate well, and presumably this effort could have been more productively spent elsewhere. This can be seen as a type of tragedy of the commons; everyone would be better off if nobody spent effort on manipulation, but individually voters are still better off manipulating.

We would point out that very recently, in the New York Mayor Democratic Primary elections (the method used is IRV), attempts were made to show Zohran Mamdani as less popular and hence, convince voters to not waste votes on him or rank him lower. This fortunately didn't work and Mamdani won the nomination.

We are interested in these forms of control as they are quite common. From the Bush elections where Nader was a spoiler candidate to Miller (2008) showing that enfranchisement of woman voters (in many nations) was beneficial for candidates who wanted to increase spending on Infant Health.

# Why does all of this matter? (iv)

We also see this in the papal conclave where Pope Paul VI brought the rule that cardinals above 80 may not participate in the conclave; effectively disenfranchising a lot of the conservative cardinals appointed by Pope John XXIII and Pius XII. We also saw this in the 2025 papal conclave where Pope Francis controlled the election from beyond the veil as he appointed 110 cardinals (of the 135 candidates and voters).

- There are also some papers in Political economics side like Nichter, 2008 which talk about turnout buying.
- In secret ballots, Finan and Schechter, 2012 explore the effects of bribery.
- The hidden failure of panchayat system due to control and bribery is explored in Anderson, François and Kotwal (2011).

Lobbying problem (studied by Christian et al. (2007) and later on by Bredereck et al. (2014a) and Binkele-Raible et al. (2014)): We are given a collection of yes/no votes over all items independently, where an item is accepted with a simple majority of yes votes, and is rejected otherwise. The

# Why does all of this matter? (v)

lobby's goal is to change the outcome to its liking by bribing certain voters without exceeding its budget.

- I do not remember the source, it is either Dutta, Watson or Dixit's text on Game Theory where there is a drawn out example on multiple rounds of lobbying; similar to bribing the lower, then upper house and then the president.
- The commonly cited political economics paper is Grossman and Helpman (1994) and competing lobbies in Groseclose and Snyder (1996).

# Why voting rules matter?

> All governments suffer a recurring problem: Power attracts pathological personalities. It is not that power corrupts but that it is magnetic to the corruptible.
>
> — Frank Herbert

There is a presentation by Ricardo Visinho, LSE and others titled “The Role of Sortition in Student Democracy” where we saw that by changing the voting rule, we were able to avoid power grabs by the usual kind of loud power-hungry candidates plurality attracts and instead get more meek but well-meaning candidates.

We also are seeing this in New York where candidates have the option to run the mayoral race on principal and not go to the lowest common denominator as the voting system is STV and that does make a difference.

# Conclusion

When voting theory started, it was an branch of political science. Once money and power entered, it became a branch of economics. And once we realized the size of nations, it became a problem of computer science.

## What we saw?

- Some famous voting rules.
- How computational hardness can help better analyze voting rules.
- We have seen some famous hardness proofs and some famous manipulation algorithms.

What we didn't see?

- A lot of voting rules like Ranked Pairs, Black, k-Approval, Maxim, $\alpha$-Condorcet, D21 etc.
- Communication Complexity of Voting Rules aka how much does the voter need to think

# Conclusion (ii)

- Parametrized and Approximation Algorithms for the hardness. For example, Borda's coalition manipuation can be approximated to $1 +$optimal number of manipulators. Also, the winner of Kemeny can be parameterized to average KT distance.
- A few other types of control. We didn't see destructive control (where we don't want a particular candidate to win) or agenda control.
- A lots of types of bribery as I mentioned in examples.
- Multimodal attacks where control and bribery are happening together.
- Adversarial Attacks where control and bribery are opposing each other.
- Voting on restricted domains (not all prefrences make sense, after all partisanship is a thing!)
- Game Theoretic voting and Voting against incomplete information and Candidate Voter Modals.
- Voting on combinatorial domains like matchings or division etc. An example is the loop that a plurality vote makes when 3 mayors divide a city fund.

# Conclusion (iii)

- and so much more.

Want to guess why we didn't see a lot of these? One, because I have in the two student seminars I have given built a reputation for going over time and I want to fix that. Two, as a lot of this is open.

As of writing this, we got optimal approximation algorithms for approval or Condorcet voting via Covering Integer Programs (CIPs) as well as give an O(m)-approximation algorithm for plurality, and a lower bound $\Omega\left(m^{\frac{1}{4}}\right)$ via Minimum k-Union (MkU) problem[2]. This was a problem we studied today, in what was for all practical purposes an introduction to the theory of voting![3]

**Thanks for listening to me yap and I shall now yap about any questions you have!**

[2]This was the first application of MkU in computational social choice

[3]Bui, Chavrimootoo, Le 2025 is the reference and is surprisingly readable.

# Appendix A

# Lemma 1

***Proof of Lemma 1.*** As is the case with the previous two theorems, we can prove something stronger.

**Definition 21:** Quasi-Condorcet.

Quasi-Condorcet is a weaker form of Condorcet where the winner is atleast quasi-Condorcet.

$a_i$ is a quasi-Condorcet alternative if $(\ldots, a_j, a_i, \ldots) \in f(A) \Longrightarrow (\ldots, a_i, a_j, \ldots) \in f(A)$. That is we are okay with swaps as long as both directions are there.

**Lemma 4.**

The only voting system that satisfies
(a) neutrality
(b) consistency
(c) Quasi-Condorcet winner
is Kemeny Rule, Anti-Kemeny Rule and Trivial function.

# Lemma 1 (ii)

Where

$$(c_1, c_2, ..., c_n) \in K(x) \Longleftrightarrow -K(x) = (c_n, ..., c_2, c_1)$$

and

$$T(x) = \mathrm{perm}(n)$$

# Lemma 3

***Proof of Lemma 3.*** It is easy to see[4] that Neutrality and Consistency imply that our voting rule is only interested in the matrix $M = m_{i,j}$ where

$$m_{i,j} = \#\text{ voters for whoom } c_i \succ c_j - \#\text{ voters for whoom } c_j \succ c_i.$$

This makes $M$ skew symmetric and also a representation of a tournament graph.

We now proceed with a proof by induction.

[4]Although hard to prove

# Lemma 3 (ii)

## $m = 2$

It is clear that we only need to consider a single graph and single case and we can see our solutions work and are the only ones that work.

## $m = 3$

Consistency guarantees that we just need to find the basis (positive, rational, linear combination) for skew symmetric matrices such that WLOG $(A, B, C) \in K(X)$ where $X$ is the matrix. This implies in

$$X = \begin{pmatrix} 0 & x_{12} & x_{13} \\ -x_{12} & 0 & x_{23} \\ -x_{13} & -x_{23} & 0 \end{pmatrix}$$

such that

# Lemma 3 (iii)

$$\begin{aligned} x_{12} + x_{13} &\geq 0 \\ x_{12} + x_{23} &\geq 0 \\ x_{13} + x_{23} &\geq 0 \end{aligned}$$

The solutions to this have the basis

$$X = \left\{ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}; \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}; \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}; \begin{pmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix} \right\}$$

We can now check the winners for all of these and get that $K$ only works. A similar process will work for $-K$ and $T$.

# Lemma 3 (iv)

## $m \geq 4$

By consistency, $f^{-1}(\sigma) = \{x \mid \sigma \in f(x)\}$ is convex for all $\sigma$. We will now sketch the proof (filling the details is to some extent busy work and to some extent the actual work of the proof):

- Show that the interiors of the convex sets $f^{-1}(\sigma)$ and $f^{-1}(\tau)$ are disjoint for all $\sigma, \tau$.
- Consider the case where $\sigma$ and $\tau$ are neighbors. It is sufficient(by neutrality) to consider the case where is the identity $e$.
- As the sets are convex, we can find a hyper plane separating $\sigma$ and $e$.[5]
- Consider some matrices similar to the 3 case and figure out the orientation of this separating wall. The induction hypothesis is useful for this step.
- We can show that this wall is same as for $K$, this would imply $f(X) \subseteq K(X)$. The induction hypothesis is useful for this step.

[5] This is called the Hyperplane separation theorem and is hard to prove, despite seeming almost obvious.

# Lemma 3 (v)

- We show by contradiction that that there is no $a \in K(X)$ such that $a \notin f(X)$.

This proves our lemma. $\square$

By imposing Condorcet, Lemma 3 implies Lemma 1. $\square$.

# Lemma 2

***Proof of Lemma 2.*** We prove this by showing reduction from Feedback Arc Set.

**Definition 22:** Feedback Arc Set.

**Input:** Directed graph $G$, with vertices $V$ and positive integer $K$.
**Question:** Is there a subset of no more than $K$ arcs which includes at least one arc from every cycle in $G$?

Given an instance of feedback arc set, we can move to a voting scenario with $O(\text{poly}(|V|))$ voters and $V$ acting as alternatives where if $(i, j) \in G$ then $\frac{|V| +2}{2}$ voters will vote for $i$ over $j$ and $\frac{|V|-2}{2}$ voters will vote for $j$ over $i$.

It is mere bookkeeping to show that if we can get a consensus with Kemeny score less than, equal to

$$\frac{|V| \ (|V| - 1)(|V| - 2)}{2} + 4K$$

# Lemma 2 (ii)

then we have a feedback arc and not otherwise. □

# Appendix B

# Proof of NP Hardness of Copeland

**Proof**. We will show a reduction from 3,4-SAT (Every clause has 3 variables and every variable appears in 4 clauses). Note, this implies the number of clauses is even.

Given $C_1, C_2, ..., C_m$ as our clauses and $X_1, X_2, ..., X_n$ as our variables; we make a tournament with candidates

- Clauses: $C_i (i = 1...m)$
- literals $X_i, \neg X_j (j = 1...m)$
- fillers: $f_k (i = 1...30m)$
- distinguished candidate $C_0$
- balencing contestent $b$

We orient our edges as

- Each litral defeats all clauses expect the ones containing it.
- Arrange $C_i (0 \leq i \leq m)$(aka clauses and distinguished candidate) on a roundtable with $2(m + 1)$ seats in alternating manner such that the seat

# Proof of NP Hardness of Copeland (ii)

diametrically opposite is empty. Let $C_i$ defeat the people on his right and lose to people on his left.

- We do the same scheme with litrals. This time such that $X_i$ and $\neg X_i$ are diametrically opposed. $X_i$ defeats people on their right and loses to people on their left. The match between $X_i, \neg X_i$ will be talked about later.
- Arrange the fillers on a roundtable with the order $f_1, f_3, ... f_{30m-1}, f_2, f_4, ..., f_{30m}$. Each filler defeats $15m - 1$ fillers to it's right and loses to $15m - 1$ filers to it's left. In the diametrically opposed fillers, the even defeats the odd.
- Every filler defeats every literal
- $C_0$ defeats $X_1$ and $\neg X_1$ but loses to $X_j$ and $\neg X_j$ for $2 \leq j \leq n$.
- $C_0$ defeats $b$
- $b$ defeats all clauses.
- $b$ loses to all litrals
- $b$ loses to all but the last $n + 4$ fillers.

# Proof of NP Hardness of Copeland (iii)

- For $C_i (0 \leq i \leq m)$, define $f_{26i+1}, f_{26i+2}, \ldots, f_{26i+26}$ to be associates of $C_i$. Thus, $C_i$[6] has 26 associates, no filler is an associate of more than one $C_i$ and nearly $4m$ fillers are not associates. $C_i$ defeats all it's non-associates and 13 of it's associates (choose the even indexed ones).

Note that we have left edges between $X_i$ and $\neg X_i$ undecided. Let the assignment of these be $R$.

**Lemma 5.**

For any any $R$, $C_i (0 \leq i \leq m)$ are tied to win under the Copeland rule.

**Proof**

$$S(C_i)(0 \leq i \leq m) = \frac{m}{2} + 3 + 0 + (30m - 13)$$
$$= \frac{m}{2} + 2 + 1 + (30m - 13) = S(C_0)$$

[6]The paper has an typo as C_1 is written here which is plain wrong.

# Proof of NP Hardness of Copeland (iv)

where the victories are over $C_i (0 \leq i \leq m)$, litrals, $B$ and fillers.

The graph has $1 + m + 2n + 30m + 1 = 31m + 2n + 2 = 31m + 3\frac{m}{2} + 2 = 65\frac{m}{2} + 2$ nodes. $B$ loses to $26m$ fillers, each filler loses to atleast $14m$ fillers, each litral loses to atleast 30m fillers; hence for all $v \neq c_i$, $S(v) \leq S(C_i)$.■

**Lemma 6.**

The second order Copeland score of $C_0$ is is independent of $R$.

**Proof** The only first order copeland scores we don't know are of the litrals (where the only thing we don't know is the result between $X_i$ and $\neg X_i$). But as $C_0$ defeats both $X_1$ and $\neg X_1$, we will be left with a constant $S^2(C_0)$ independent of $R$. ■

**Lemma 7.**

For all non-clause candidates, second order Copeland scores are less than $S^2(C_0)$ for any $R$.

# Proof of NP Hardness of Copeland (v)

**Proof** For any $C_i$, the portion of second order score comming from fillers is about $30m * (15m + 2n) > 480m^2$. As in Lemma 1 of the proof, every other nodes loses to atleast $14m$ fillers. Hence, the filler portion of other second order scores will be smaller by atleast $14m(15m + 2) > 225m^2$. This is a huge deficit which can't be covered by the remaining $m$ nodes, which can atmost contribute $16m$ each. ■

**Lemma 8.**

The second order copeland scores of $C_i(1 \leq i \leq m)$ ignoring the undecided edges is $S^2(C_0) - 3$.

Let this tentative second order copeland score be reprasented by $T^2(C_i)$. We will first try to show that $T^2(C_i) = T^2(C_{i+1})$ and hence, $T^2(C_1) = T^2(C_2) = ... = T^2(C_m)$.

Define $L(C_i)$ be the total number (counting repetitions) of clauses of which the literals in $C_i$ are members, plus 1 if $C_i$ contains $X_1$ or $\neg X_1$. We count $C_0$

# Proof of NP Hardness of Copeland (vi)

as containing $X_1$ and $\neg X_1$. Thus, $L(C_i)$ is the number of contests won by one of $C_j (0 \leq j \leq m)$, against the three literals that comprise $C_i$.

Consider $T^2(C_i) - T^{C_{i+1}} = D_C + D_B + D_L + D_F$ where $D$ reprasents difference due to clauses, balence, litrals and fillers. Notice,

- $D_C = 0$ as both beat $\frac{m}{2}$ clauses.
- $D_B = 0$ as both beat $B$.
- $D_L = 3(m + n + 1) - L(C_i) - [3(m + n + 1) - L(C_{i+1})] = L(C_{i+1}) - L(C_i)$
- $D_F = 0$ from non associate fillers $+ L(C_i) - L(C_{i+1}) + 13 - 13 = L(C_i) - L(C_{i+1})$

Thus, $T(C_i) = T(C_i)$.

Now we do the same procedure on $S^2(C_0) - T(C_1)$

- $D_C = 0$ as both beat $\frac{m}{2}$ clauses.
- $D_B = m + n + 4$ as $C_0$ beats $B$.

# Proof of NP Hardness of Copeland (vii)

- $D_L = 2(n+m+1) - L(C_0) - 3(n+m+1) + L(C_1) = -(n+m+1) + L(C_1) - L(C_0)$[7]
- $D_F = l(C_0) - L(C_1)$

Thus, the difference is $m + n + 4 - n - m - 1 = 3$. ■

To finally finish this proof, we claim that a $R$ which make $c$ the sole winner only exist if and only if the 3,4-SAT is satisfiable.

Imagine candidate $C_1 (i = l \ldots m)$ during the last round of contests. His own contests are over, so his second order Copeland score has been determined, and by lemma 4, is currently 3 points short of a share of first place with $C_0$ and possibly other $C_j$.

By construction, $C_i$ has lost to all but 3 of the literals, so the outcomes of only 3 contests (those containing the literals defeated by $C_i$) could improve his

[7] There is a typo in the sign here, in the orignal paper!

# Proof of NP Hardness of Copeland (viii)

second-order score. If all 3 contests go as $C_i$ wishes (the candidates that $C_i$ defeated win their contests), then $C_{i'}s$ second-order score will equal that of $C_0$.

Interpreting a literal losing to its complement as the literal being set to TRUE in the instance of 3,4-SAT, candidate $C_i$ will lose to $c$ if and only if clause $C_i$ is satisfied. Thus satisfiability of the 3,4-SAT expression corresponds precisely to all of the $C_i$'s ($i = ...m$) being defeated by $C_0$.

We can create a set of voter's $V$ which realizes this graph such that every edge other than between $X_i$ and $\neg X_i$ is decided by 2 votes and these are tied. Thus, the mnaiuplater can only decide $R$ which is NP-complete. ■