Computational social choice

Alexandros Voudouris

University of Oxford

Making decisions

- Alice, Bob and Carol want to decide where to go for dinner

Making decisions

- Alice, Bob and Carol want to decide where to go for dinner
- There are three restaurant options: Franco Manca, White Rabbit, Zizzi

Making decisions

- Alice, Bob and Carol want to decide where to go for dinner
- There are three restaurant options: Franco Manca, White Rabbit, Zizzi

- Each of the friends has preferences over the restaurants:
  - Alice prefers Franco Manca the most, and White Rabbit to Zizzi
  - Bob prefers White Rabbit the most, and Zizzi to Franco Manca
  - Carol prefers Franco Manca the most, and White Rabbit to Zizzi

Making decisions

- Alice, Bob and Carol want to decide where to go for dinner
- There are three restaurant options: Franco Manca, White Rabbit, Zizzi
- Each of the friends has preferences over the restaurants:
  - Alice prefers Franco Manca the most, and White Rabbit to Zizzi
  - Bob prefers White Rabbit the most, and Zizzi to Franco Manca
  - Carol prefers Franco Manca the most, and White Rabbit to Zizzi
- How should they decide where to go?

Making decisions

- Alice, Bob and Carol want to decide where to go for dinner
- There are three restaurant options: Franco Manca, White Rabbit, Zizzi
- Each of the friends has preferences over the restaurants:
  - Alice prefers Franco Manca the most, and White Rabbit to Zizzi
  - Bob prefers White Rabbit the most, and Zizzi to Franco Manca
  - Carol prefers Franco Manca the most, and White Rabbit to Zizzi
- How should they decide where to go?
- They can vote!

Making decisions

- There are many ways to vote however

Making decisions

- There are many ways to vote however
- One way is for everyone to vote only for their favourite restaurant, and then choose the restaurant with the most votes:
  - Alice and Carol vote Franco Manca, and Bob votes White Rabbit
  - Franco Manca is chosen

Making decisions

- There are many ways to vote however
- One way is for everyone to vote only for their favourite restaurant, and then choose the restaurant with the most votes:
  - Alice and Carol vote Franco Manca, and Bob votes White Rabbit
  - Franco Manca is chosen
- But, observe that Bob really doesn't like Franco Manca

Making decisions

- There are many ways to vote however
- One way is for everyone to vote only for their favourite restaurant, and then choose the restaurant with the most votes:
  - Alice and Carol vote Franco Manca, and Bob votes White Rabbit
  - Franco Manca is chosen
- But, observe that Bob really doesn't like Franco Manca
- Another way is for everyone to veto their most disliked restaurant, and then choose the restaurant with the least vetos
  - Alice and Carol veto Zizzi, and Bob vetos Franco Manca
  - White Rabbit is chosen

Making decisions

- One more way is to count for each restaurant the number of restaurants it beats in pairwise comparisons, and then choose the restaurant with the most wins:
  - Franco Manca beats both White Rabbit and Zizzi twice
  - White Rabbit beats Franco Manca once, and Zizzi three times
  - Zizzi beats only Franco Manca once

Making decisions

- One more way is to count for each restaurant the number of restaurants it beats in pairwise comparisons, and then choose the restaurant with the most wins:
  - Franco Manca beats both White Rabbit and Zizzi twice
  - White Rabbit beats Franco Manca once, and Zizzi three times
  - Zizzi beats only Franco Manca once
- Franco Manca and White Rabbit have 4 wins each

Making decisions

- One more way is to count for each restaurant the number of restaurants it beats in pairwise comparisons, and then choose the restaurant with the most wins:
  - Franco Manca beats both White Rabbit and Zizzi twice
  - White Rabbit beats Franco Manca once, and Zizzi three times
  - Zizzi beats only Franco Manca once
- Franco Manca and White Rabbit have 4 wins each

- The decision depends on how this tie is broken
- For example, using the pairwise comparison between these two restaurants, Franco Manca is finally chosen

Our setting

- A set of n **agents**: $N = \{1,2, \dots , n\}$
- A set of m **alternatives**: $A = \{a_1, a_2, \dots , a_m\}$

Our setting

- A set of n **agents**: $N = \{1,2, \dots, n\}$
- A set of m **alternatives**: $A = \{a_1, a_2, \dots, a_m\}$

- Every agent has **preferences over the alternatives** and provides an **ordering** (**ranking**) of them

Our setting

- A set of n **agents**: $N = \{1,2,\dots,n\}$
- A set of m **alternatives**: $A = \{a_1, a_2, \dots, a_m\}$

- Every agent has **preferences over the alternatives** and provides an **ordering** (**ranking**) of them

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

Our setting

- A set of n **agents**: $N = \{1,2,\ldots,n\}$
- A set of m **alternatives**: $A = \{a_1, a_2, \ldots, a_m\}$

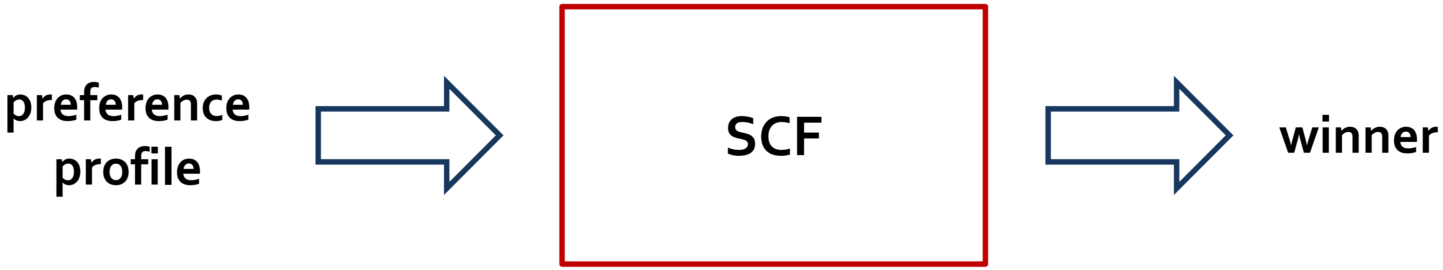
- Every agent has **preferences over the alternatives** and provides an **ordering** (**ranking**) of them

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

- Our **goal** is to select an alternative or come up with a ranking over all alternatives, by taking into account the preferences of the agents

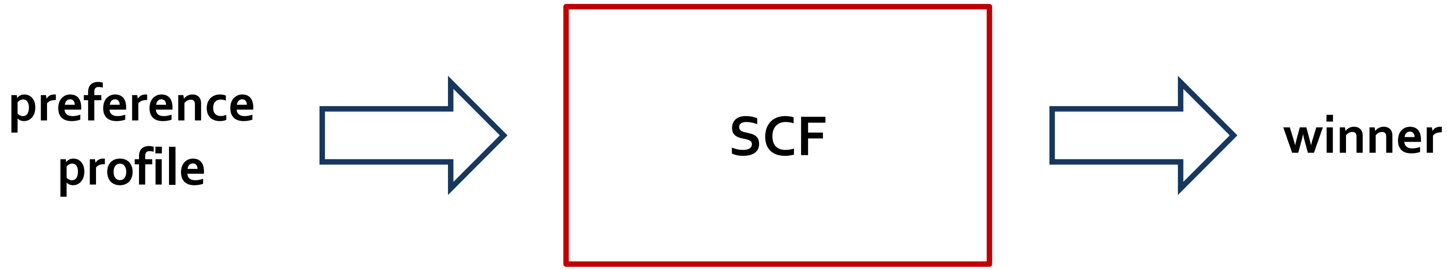
Social choice and welfare functions

- A social choice function (SCF) takes as input a preference profile, and outputs a winning alternative

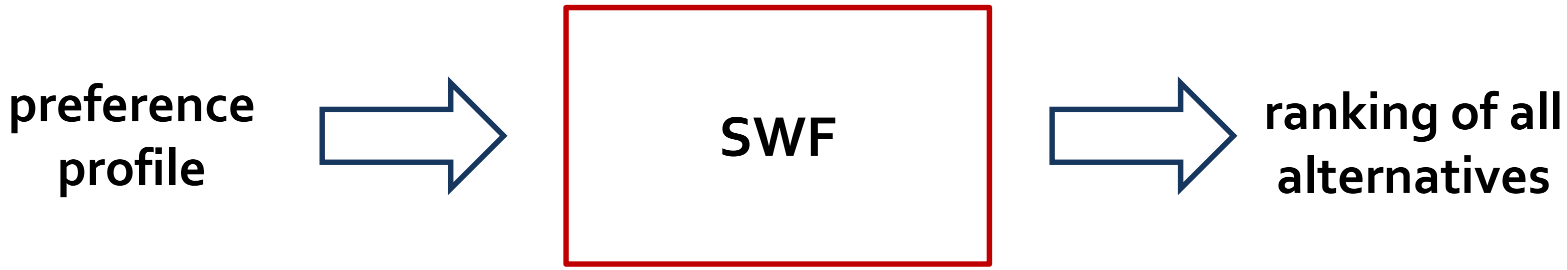


Social choice and welfare functions

- A social choice function (SCF) takes as input a preference profile, and outputs a winning alternative



- A social welfare function (SWF) takes as input a preference profile, and outputs a complete ranking of all alternatives



Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	**ranking**			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

$\boldsymbol{s}$	4	2	1	0

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \ldots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

$\boldsymbol{s}$	4	2	1	0

alternative	points
a	0
b	0
c	0
d	0

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	$\boldsymbol{b}$	d	a	c
2	$\boldsymbol{d}$	a	c	b
3	$\boldsymbol{d}$	c	a	b
4	$\boldsymbol{a}$	b	c	d

$\boldsymbol{s}$	**4**	2	1	0

alternative	points
a	4
b	4
c	0
d	8

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	$\boldsymbol{d}$	a	c
2	d	$\boldsymbol{a}$	c	b
3	d	$\boldsymbol{c}$	a	b
4	a	$\boldsymbol{b}$	c	d
$\boldsymbol{s}$	4	**2**	1	0

alternative	points
a	6
b	6
c	2
d	10

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	d	$\boldsymbol{a}$	c
2	d	a	$\boldsymbol{c}$	b
3	d	c	$\boldsymbol{a}$	b
4	a	b	$\boldsymbol{c}$	d

$\boldsymbol{s}$	4	2	**1**	0

alternative	points
a	8
b	6
c	4
d	10

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	d	a	$\boldsymbol{c}$
2	d	a	c	$\boldsymbol{b}$
3	d	c	a	$\boldsymbol{b}$
4	a	b	c	$\boldsymbol{d}$

$\boldsymbol{s}$	4	2	1	**0**

alternative	points
a	8
b	6
c	4
d	10

Positional scoring rules

- A PSR is defined by a scoring vector of size m: $\boldsymbol{s} = (s_1, s_2, \dots, s_m)$
- For every agent, the alternative that is ranked k-th gets s_k points
- The alternatives are ranked according to their total points

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d
$\boldsymbol{s}$	4	2	1	0

alternative	points	
a	8	
b	6	
c	4	
d	10	winner!

Positional scoring rules

- **Plurality:** give a point to the favourite alternative of each agent, and rank the alternatives in terms of total score
  - $\mathbf{PL} = (1,0, \ldots, 0, 0)$

Positional scoring rules

- **Plurality:** give a point to the favourite alternative of each agent, and rank the alternatives in terms of total score
  - $\mathbf{PL} = (1,0, \dots, 0, 0)$

- **Veto:** for every agent give a point to every alternative besides the least favourite alternative of the agent, and rank the alternatives in terms of total score
  - $\mathbf{VE} = (1,1, \dots, 1, 0)$

Positional scoring rules

- **Plurality:** give a point to the favourite alternative of each agent, and rank the alternatives in terms of total score
  - $\mathbf{PL} = (1,0, \dots, 0, 0)$

- **Veto:** for every agent give a point to every alternative besides the least favourite alternative of the agent, and rank the alternatives in terms of total score
  - $\mathbf{VE} = (1,1, \dots, 1, 0)$

- **Borda:** give a point to an alternative for every pairwise win against another alternative, and rank the alternatives in terms of total score
  - $\mathbf{B} = (m-1, m-2, \dots, 1, 0)$

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	**ranking**			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

alternative	**points**
a	0
b	0
c	0
d	0

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	ranking			
1	$\boldsymbol{b}$	d	$\boldsymbol{a}$	c
2	d	$\boldsymbol{a}$	c	$\boldsymbol{b}$
3	d	c	$\boldsymbol{a}$	$\boldsymbol{b}$
4	$\boldsymbol{a}$	$\boldsymbol{b}$	c	d

alternative	points
a	1
b	0
c	0
d	0

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

alternative	points
a	2
b	0
c	0
d	0

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

alternative	points
a	2
b	0
c	0
d	1

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

alternative	points
a	2
b	1
c	0.5
d	2.5

Copeland

- Similarly to PSRs, every alternative has a score and the winner is the alternative with the highest score
- We say that an alternative x pairwise beats another alternative y if the majority of agents prefer x to y
- The score of an alternative x is equal to the number of alternatives that x pairwise beats, plus half the number of alternatives that pairwise ties

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	c	a	b
4	a	b	c	d

alternative	points	
a	2	
b	1	
c	0.5	
d	2.5	**winner!**

Ranked pairs

- We first create a ranking of all ordered pairs of alternatives, by sorting them in terms of the number of pairwise victories, breaking ties arbitrarily

Ranked pairs

- We first create a ranking of all ordered pairs of alternatives, by sorting them in terms of the number of pairwise victories, breaking ties arbitrarily
- Starting from the top pair according to this ranking, we lock the relative order of the next pair of alternatives if and only if it satisfies the ranking that has been created so far

Ranked pairs

- We first create a ranking of all ordered pairs of alternatives, by sorting them in terms of the number of pairwise victories, breaking ties arbitrarily
- Starting from the top pair according to this ranking, we lock the relative order of the next pair of alternatives if and only if it satisfies the ranking that has been created so far
- We can model the execution of this process by a directed graph, where each node represents an alternative and an edge from some alternative x to an alternative y represents the fact that x is ranked higher than y

Ranked pairs

- We first create a ranking of all ordered pairs of alternatives, by sorting them in terms of the number of pairwise victories, breaking ties arbitrarily
- Starting from the top pair according to this ranking, we lock the relative order of the next pair of alternatives if and only if it satisfies the ranking that has been created so far
- We can model the execution of this process by a directed graph, where each node represents an alternative and an edge from some alternative x to an alternative y represents the fact that x is ranked higher than y
- So, we successively add edges to this graph following the ranking of pairs as long as no cycle is created

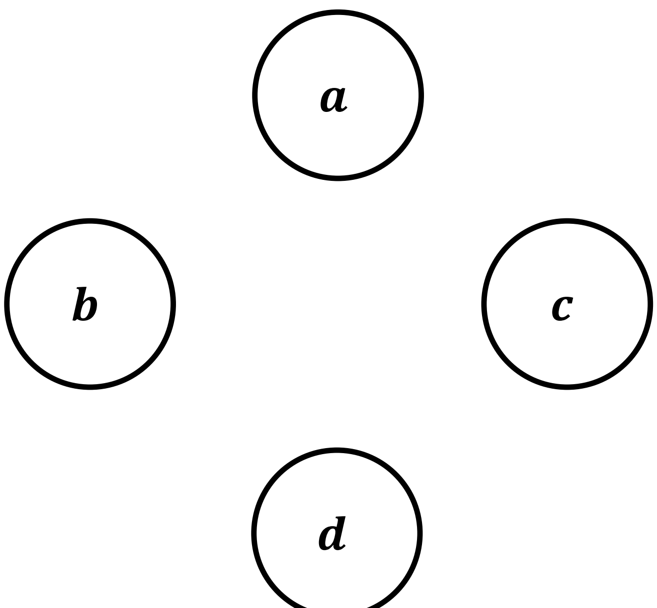
Ranked pairs

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d

pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

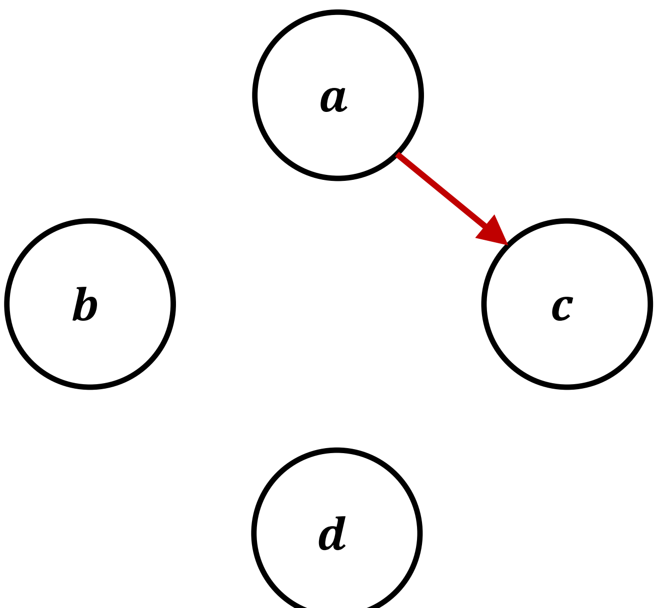
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a,c)	4
(a,b)	3
(d,c)	3
(d,a)	3
(c,b)	2
(b,d)	2
(b,c)	2
(d,b)	2
(a,d)	1
(b,a)	1
(c,d)	1
(c,a)	0

Ranked pairs

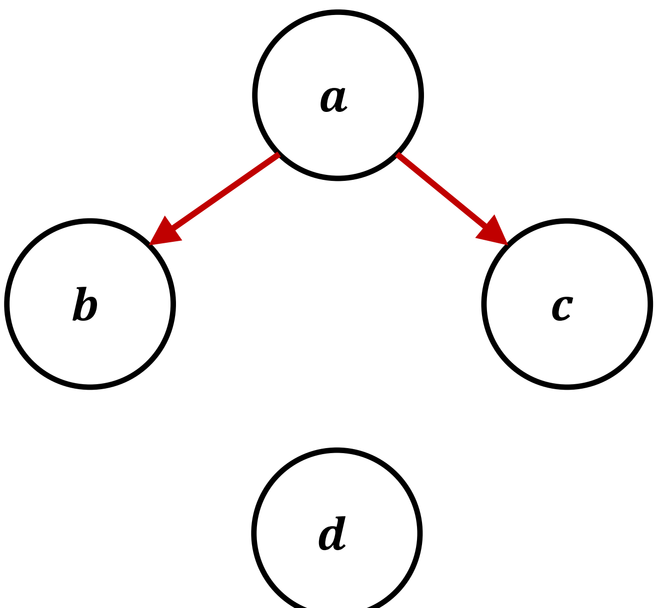
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
$(\boldsymbol{a}, \boldsymbol{c})$	**4**
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

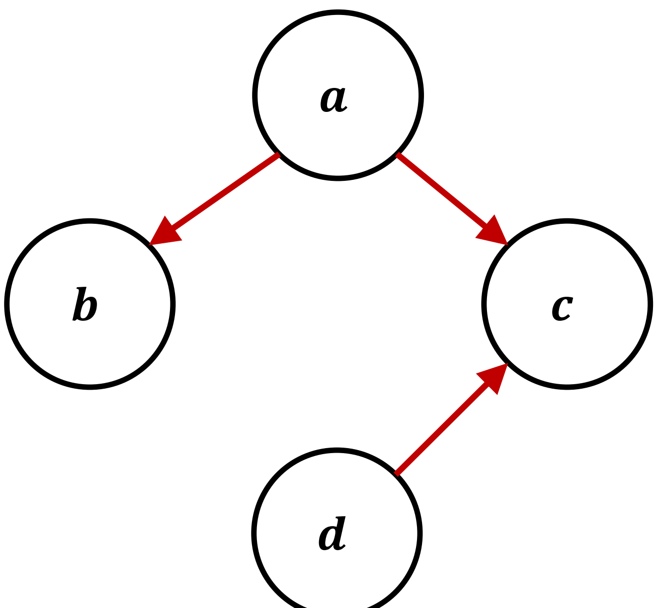
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

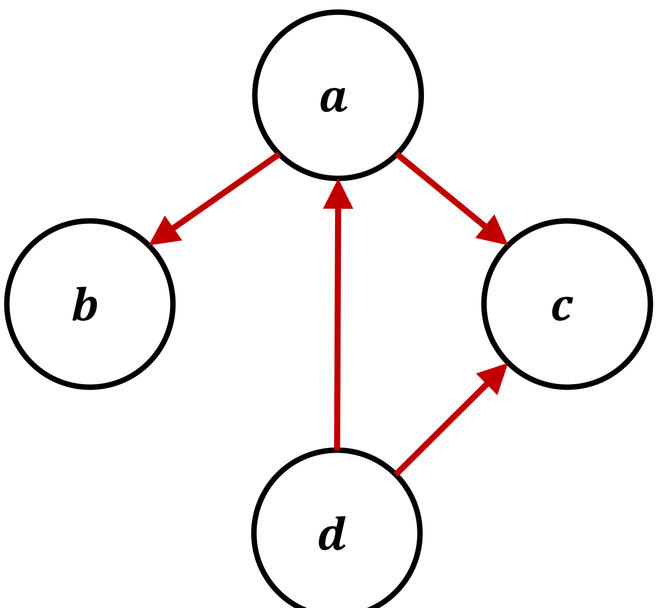
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

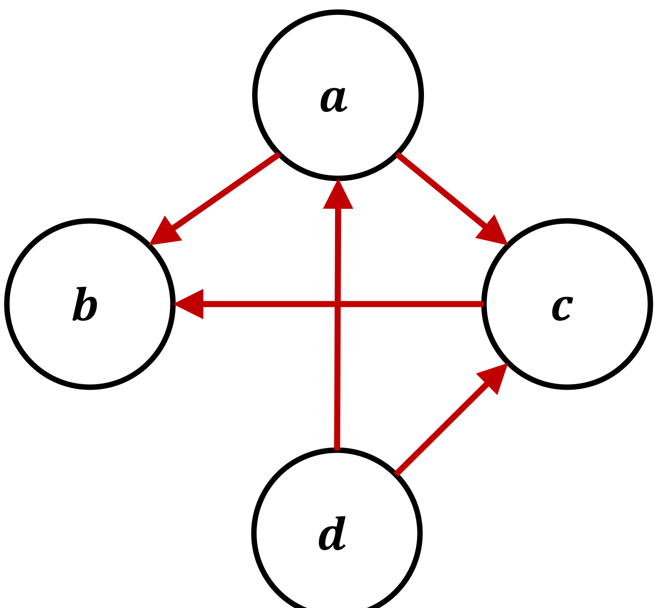
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
$\boldsymbol{(d, a)}$	**3**
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

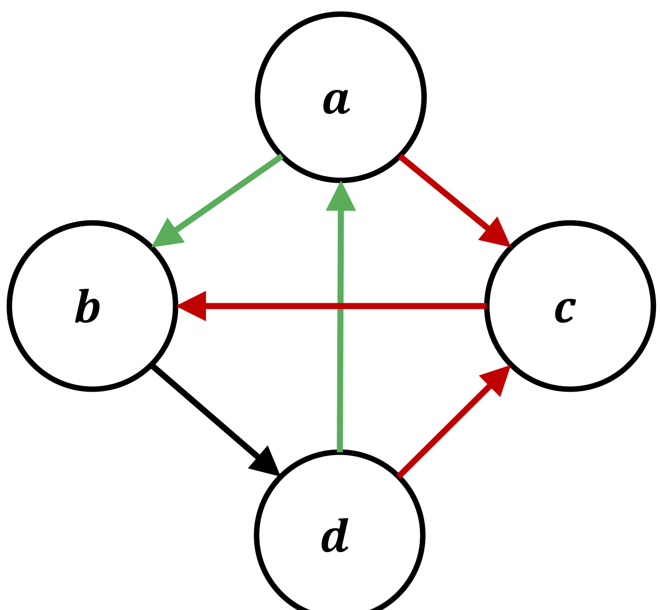
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

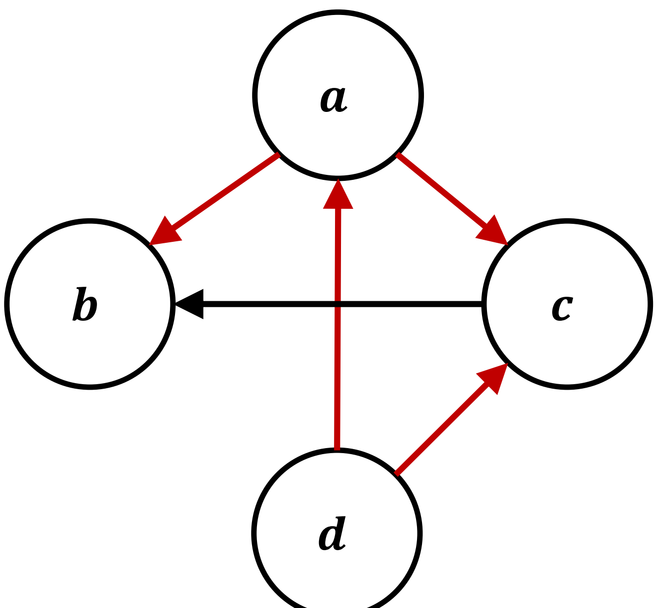
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
$\boldsymbol{(b, d)}$	**2**
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

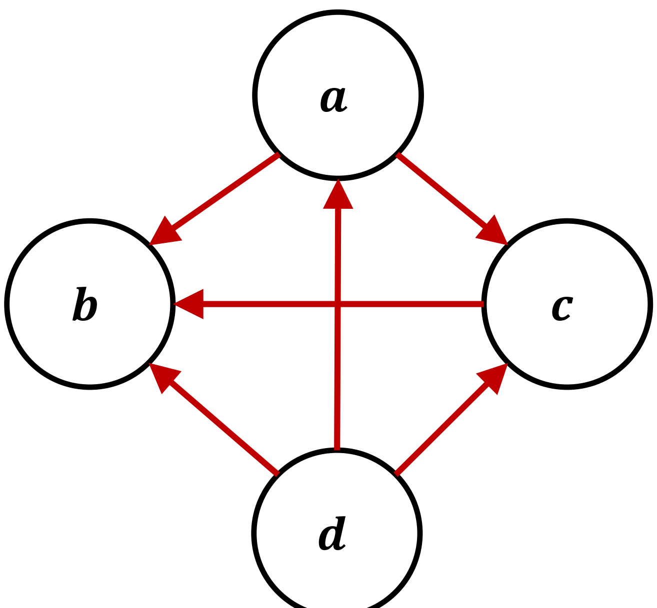
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
$\boldsymbol{(b, c)}$	**2**
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

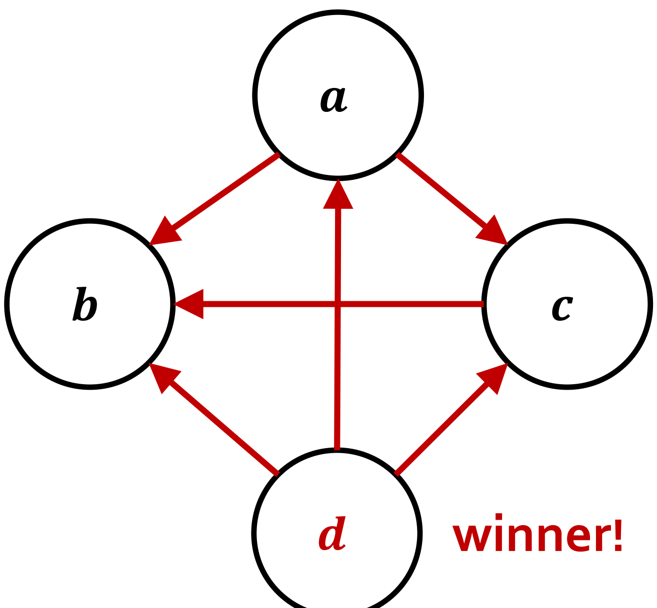
agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
$\boldsymbol{(d, b)}$	**2**
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Ranked pairs

agent	ranking			
1	b	d	a	c
2	d	a	c	b
3	d	a	c	b
4	a	b	c	d



pair	victories
(a, c)	4
(a, b)	3
(d, c)	3
(d, a)	3
(c, b)	2
(b, d)	2
(b, c)	2
(d, b)	2
(a, d)	1
(b, a)	1
(c, d)	1
(c, a)	0

Dictatorship

- The simplest and most unfair voting rule
- The output is the favourite alternative or the whole preference of a particular agent
- Naturally, this agent is called the dictator

Some desired properties

- **Unanimity:** If all agents have exactly the same preferences over the alternatives, then the output should be what everyone wants

Some desired properties

- **Unanimity:** If all agents have exactly the same preferences over the alternatives, then the output should be what everyone wants

agent	ranking			
1	a	b	c	d
2	a	b	c	d
3	a	b	c	d
4	a	b	c	d

Some desired properties

- **Independence of Irrelevant Alternatives (IIA):** the relative order of two alternatives does not depend on other alternatives

Some desired properties

- **Independence of Irrelevant Alternatives (IIA):** the relative order of two alternatives does not depend on other alternatives
  - The relative order of two alternatives x and y in the outcome ranking should be the same for all input preference profiles that consist of rankings where x and y have the same order

Some desired properties

- **Independence of Irrelevant Alternatives (IIA):** the relative order of two alternatives does not depend on other alternatives
  - The relative order of two alternatives x and y in the outcome ranking should be the same for all input preference profiles that consist of rankings where x and y have the same order

agent	ranking			
1	d	c	b	a
2	a	c	d	b
3	a	d	b	c
4	b	a	c	d

agent	ranking			
1	c	b	d	a
2	a	b	c	d
3	c	d	a	b
4	d	c	b	a

Some desired properties

- **Independence of Irrelevant Alternatives (IIA):** the relative order of two alternatives does not depend on other alternatives
  - The relative order of two alternatives x and y in the outcome ranking should be the same for all input preference profiles that consist of rankings where x and y have the same order

agent	ranking			
1	d	c	$\boldsymbol{b}$	$\boldsymbol{a}$
2	$\boldsymbol{a}$	c	d	$\boldsymbol{b}$
3	$\boldsymbol{a}$	d	$\boldsymbol{b}$	c
4	$\boldsymbol{b}$	$\boldsymbol{a}$	c	d

agent	ranking			
1	c	$\boldsymbol{b}$	d	$\boldsymbol{a}$
2	$\boldsymbol{a}$	$\boldsymbol{b}$	c	d
3	c	d	$\boldsymbol{a}$	$\boldsymbol{b}$
4	d	c	$\boldsymbol{b}$	$\boldsymbol{a}$

Some desired properties

- Unanimity and IIA seem to be two very natural properties to request from a voting rule to satisfy

Some desired properties

- Unanimity and IIA seem to be two very natural properties to request from a voting rule to satisfy
- But, …

Theorem [Arrow, 1951]
For at least three alternatives, any unanimous and IIA social welfare function must be a dictatorship

Manipulations

- So far, we have assumed that the agents behave honestly and report their true preferences over the alternatives

Manipulations

- So far, we have assumed that the agents behave honestly and report their true preferences over the alternatives
- However, it might be possible for an agent to have incentive to misreport her preferences if this leads to an outcome that she prefers more

Manipulations

- So far, we have assumed that the agents behave honestly and report their true preferences over the alternatives
- However, it might be possible for an agent to have incentive to misreport her preferences if this leads to an outcome that she prefers more

agent	ranking			
1	a	b	c	d
2	d	c	a	b
3	d	c	b	a

alternative	Borda score
a	4
b	3
c	5
$\boldsymbol{d}$	**6**

Manipulations

- So far, we have assumed that the agents behave honestly and report their true preferences over the alternatives
- However, it might be possible for an agent to have incentive to misreport her preferences if this leads to an outcome that she prefers more

agent	ranking			
1	***a***	***b***	***c***	***d***
2	*d*	*c*	*a*	*b*
3	*d*	*c*	*b*	*a*

alternative	Borda score
a	4
b	3
c	5
d	**6**

Manipulations

- So far, we have assumed that the agents behave honestly and report their true preferences over the alternatives
- However, it might be possible for an agent to have incentive to misreport her preferences if this leads to an outcome that she prefers more

agent	ranking			
1	$\boldsymbol{c}$	$\boldsymbol{a}$	$\boldsymbol{b}$	$\boldsymbol{d}$
2	d	c	a	b
3	d	c	b	a

alternative	Borda score
a	3
b	2
$\boldsymbol{c}$	**7**
d	6

More than two options

Beyond the Majority Rule

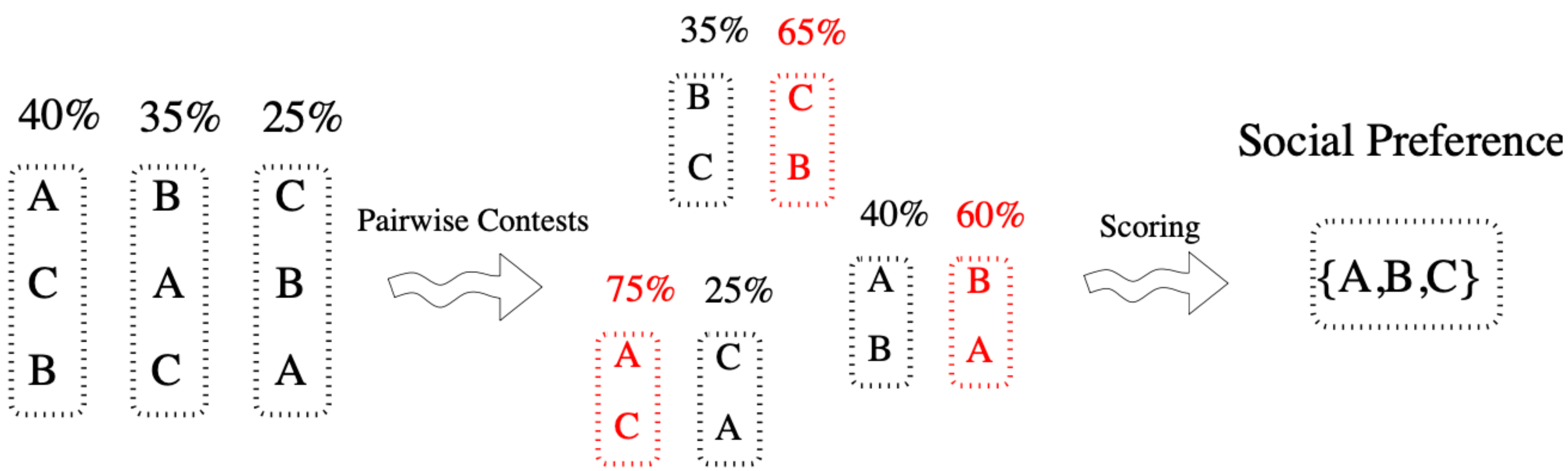


FIGURE 13.1. In pairwise contests A defeats C and C defeats B, yet B defeats A.

Plurality Voting

Choosing the favorite.

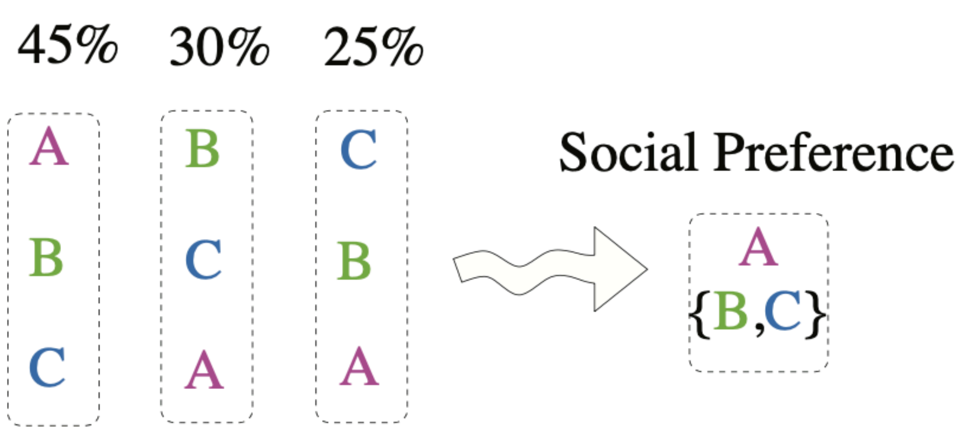


FIGURE 13.2. Option A is preferred by 45% of the population, option B by 30%, and option C by 25%, and thus A wins a plurality vote. However, A is the least favorite for 55% of the population.

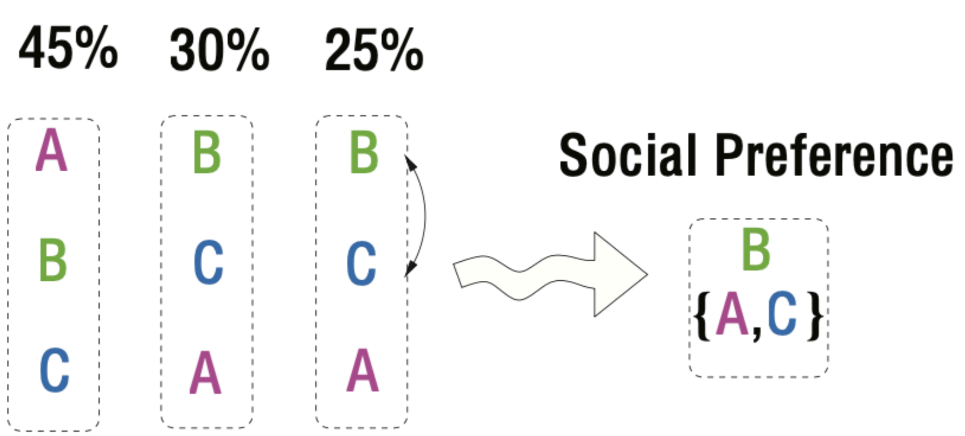


FIGURE 13.3. When 25% strategically switch their votes from C to B, the relative ranking of A and B in the outcome changes.

Runoff Elections

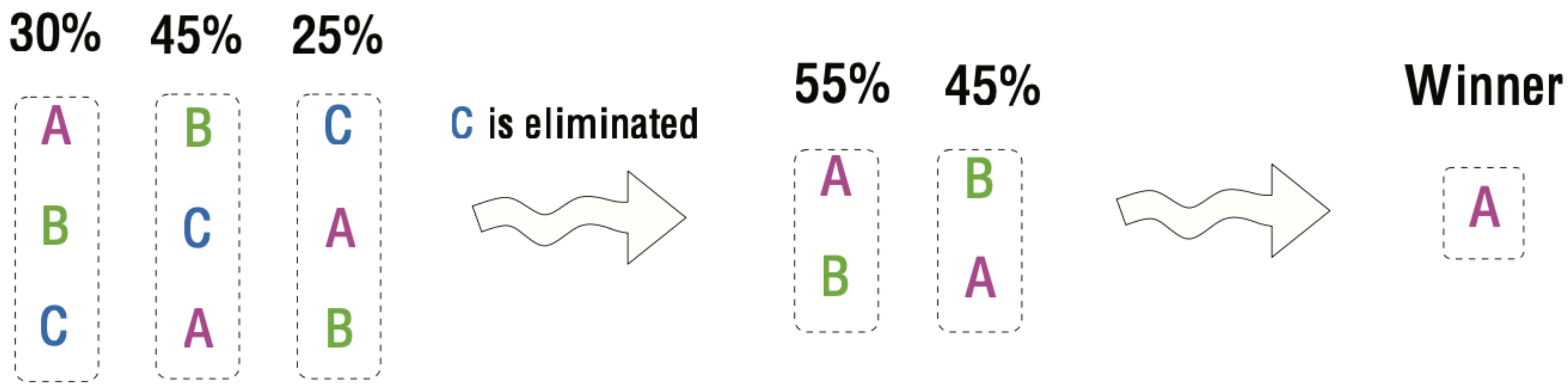


FIGURE 13.4. In the first round C is eliminated. When votes are redistributed, B gets the majority.

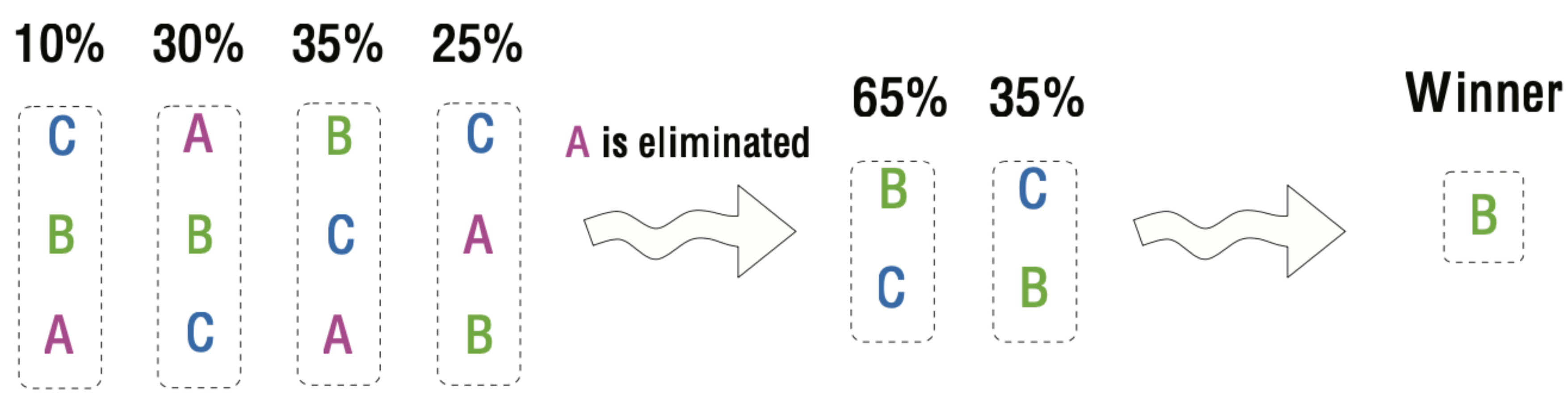


FIGURE 13.5. Some of the voters from the second group in Figure 13.4 misrepresent their true preferences, ensuring that A is eliminated. As a result, B wins the election.

Manipulations

- We would like to use voting rules that are strategy-proof, and always incentivize the agents to truthfully report their true preferences over the alternatives

Manipulations

- We would like to use voting rules that are strategy-proof, and always incentivize the agents to truthfully report their true preferences over the alternatives
- But, …

Theorem [Gibbard,1973 & Satterthwaite, 1975]
For at least three alternatives, any strategy-proof and onto the set of alternatives social choice function must be a dictatorship

Dealing with manipulations

- In general, the impossibility result of Gibbard-Satterthwaite indicates that there is now way to avoid manipulative behaviour, unless the voting rule is a dictatorship

Dealing with manipulations

- In general, the impossibility result of Gibbard-Satterthwaite indicates that there is now way to avoid manipulative behaviour, unless the voting rule is a dictatorship

- One way to “avoid” it is by using voting rules for which the problem of computing a manipulation is NP-complete

Dealing with manipulations

- In general, the impossibility result of Gibbard-Satterthwaite indicates that there is now way to avoid manipulative behaviour, unless the voting rule is a dictatorship

- One way to “avoid” it is by using voting rules for which the problem of computing a manipulation is NP-complete
- For example, some results of this flavour are as follows:
  - Computing a manipulation is easy for positional scoring rules and Copeland, but NP-complete for Ranked Pairs

Dealing with manipulations

- In general, the impossibility result of Gibbard-Satterthwaite indicates that there is now way to avoid manipulative behaviour, unless the voting rule is a dictatorship

- One way to "avoid" it is by using voting rules for which the problem of computing a manipulation is NP-complete
- For example, some results of this flavour are as follows:
  - Computing a manipulation is easy for positional scoring rules and Copeland, but NP-complete for Ranked Pairs

- Another way to "avoid" this is to focus on special cases, where the preferences of the agents are more structured

Facility location on the line

- A set of agents positioned on a line
- One **facility** to be built somewhere

Facility location on the line

- A set of agents positioned on a line
- One **facility** to be built somewhere

- Every agent has preferences over the possible locations of the facility, defined by the distance of her position from the facility: the smaller the distance, the better
  - Such preferences are called single-peaked

Facility location on the line

- A set of agents positioned on a line
- One **facility** to be built somewhere

- Every agent has preferences over the possible locations of the facility, defined by the distance of her position from the facility: the smaller the distance, the better
  - Such preferences are called single-peaked

- The agents report their positions

Facility location on the line

- A set of agents positioned on a line
- One **facility** to be built somewhere

- Every agent has preferences over the possible locations of the facility, defined by the distance of her position from the facility: the smaller the distance, the better
  - Such preferences are called single-peaked

- The agents report their positions
- The goal is to decide where to build the facility so that no agent manipulates, and without using a dictatorship

Facility location on the line

Theorem
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents

Facility location on the line

<u>Theorem</u>
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents

- The median agent has zero cost

Facility location on the line

Theorem
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents

- The median agent has zero cost
- If the blue agent reports a position smaller than the median position, nothing will change

Facility location on the line

Theorem
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents

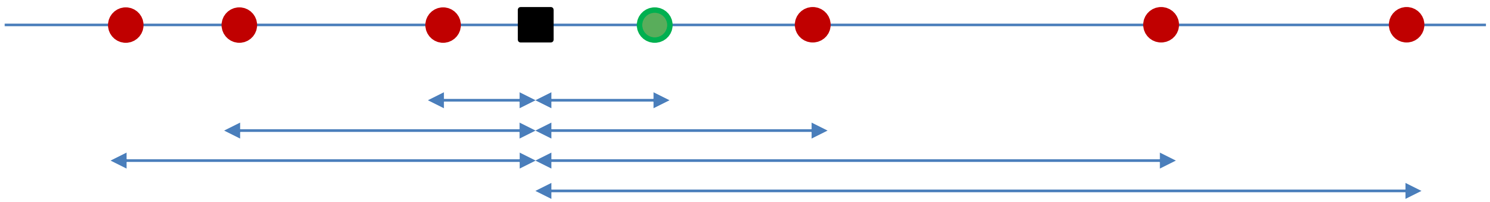
- The median agent has zero cost
- If the blue agent reports a position smaller than the median position, nothing will change
- If the blue agent reports a position larger than the median position, then the median position can only be further away from her true position

Facility location on the line

Theorem
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents

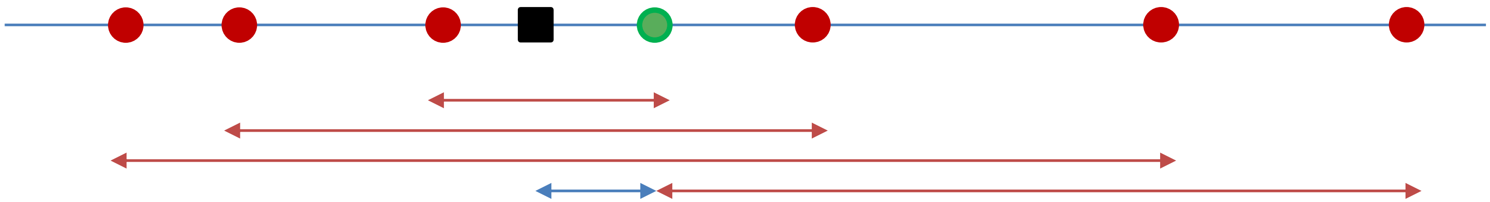
Facility location on the line

Theorem
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents



Facility location on the line

<u>Theorem</u>
Building the facility at the median agent position is strategy-proof and minimizes the total cost of the agents



Summary

Summary

- **Voting:** a way to make decisions

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative
- **Social welfare functions:** output a ranking over all alternatives

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative
- **Social welfare functions:** output a ranking over all alternatives
- Positional scoring rules, Copeland, Ranked pairs, Dictatorship

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative
- **Social welfare functions:** output a ranking over all alternatives
- Positional scoring rules, Copeland, Ranked pairs, Dictatorship
- Only dictatorship can satisfy **unanimity and independence of irrelevant alternatives** (for at least 3 alternatives)

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative
- **Social welfare functions:** output a ranking over all alternatives
- Positional scoring rules, Copeland, Ranked pairs, Dictatorship
- Only dictatorship can satisfy **unanimity and independence of irrelevant alternatives** (for at least 3 alternatives)
- Only dictatorship **cannot be manipulated** by the agents (for at least 3 alternatives)

Summary

- **Voting:** a way to make decisions
- **Social choice functions:** take as input the preferences of the agents, output a single winning alternative
- **Social welfare functions:** output a ranking over all alternatives
- Positional scoring rules, Copeland, Ranked pairs, Dictatorship
- Only dictatorship can satisfy **unanimity and independence of irrelevant alternatives** (for at least 3 alternatives)
- Only dictatorship **cannot be manipulated** by the agents (for at least 3 alternatives)
- **Facility location on the line:** selecting the median is strategy-proof and minimizes the social cost

Bibliography

- Handbook of Computational Social Choice
  - http://procaccia.info/papers/comsoc.pdf
- Trends in Computational Social Choice
  - http://research.illc.uva.nl/COST-IC1205/BookDocs/TrendsCOMSOC.pdf