Approximation algorithms and mechanism design for minimax approval voting

Ioannis Caragiannis
Dimitris Kalaitzis
University of Patras

Vangelis Markakis
Athens University of Economics and Business

Outline

- Approval Voting
  - The minisum and the minimax solution
- Approximation algorithms for the minimax solution
  - Pareto-efficiency and approximability
  - An LP-based algorithm
- Immunity to manipulation
  - Strategy-proof and group-strategyproof algorithms and lower bounds

Approval voting

- Multi-winner elections
- Every voter selects a subset of candidates that he approves of
- Used by numerous organizations (IEEE, Game Theory Society, INFORMS,…) for selecting committees

Notation:

- n voters
- m candidates
- k: size of the committee to be elected
- Votes $\Rightarrow$ elements of $\{0,1\}^m$

Electing a committee from approval ballots

m = 5 candidates

n = 4 ballots

What is the best committee of size $k = 2$?

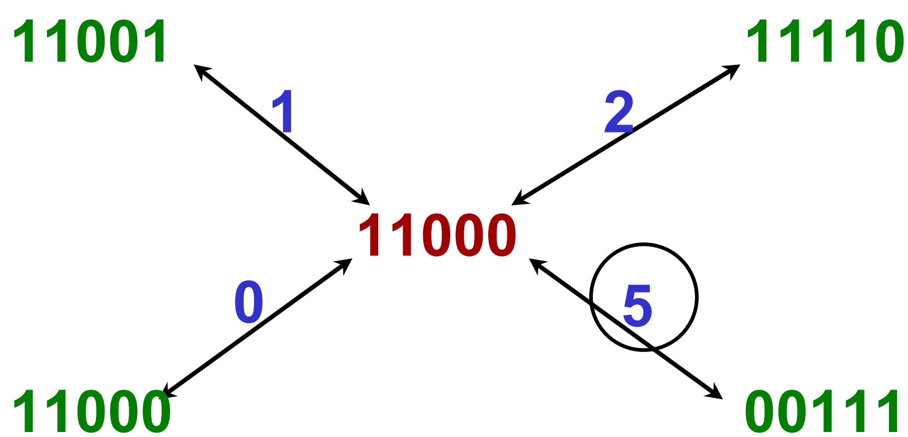
11001 **11110**

11000 **00111**

The minisum solution

Pick the committee that minimizes the sum of the Hamming distances to the voters

⇒ Pick the k candidates with the highest approval rate

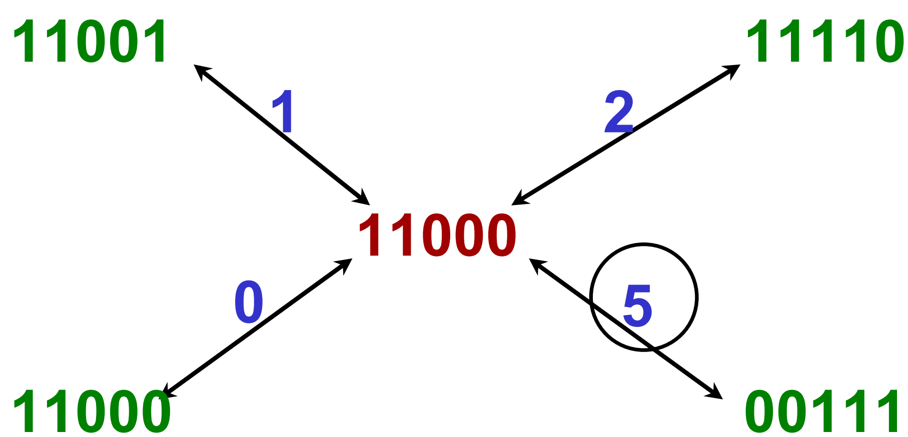


sum = 8

max = 5

Maximum Hamming distance

The preferences of some voters may be completely ignored

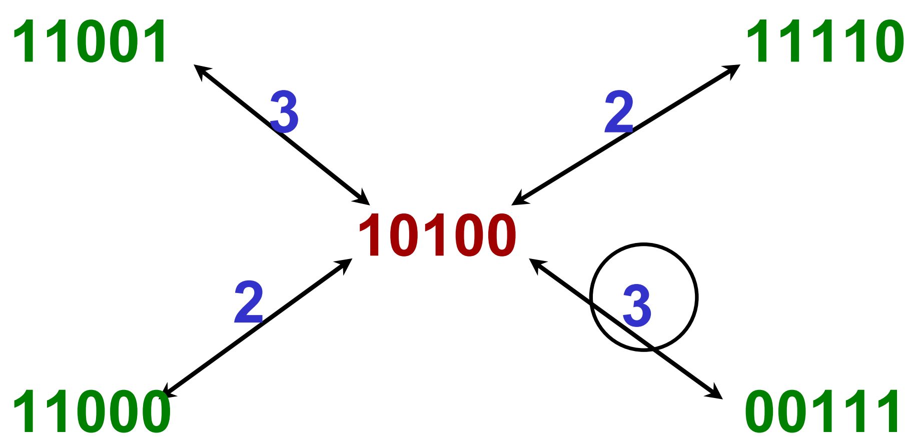


sum = 8

max = 5

The minimax solution

[Brams, Kilgour, Sanver ’07]: Pick the *k* candidates that minimize the maximum Hamming distance to a voter



sum = 10
max = 3

Some difficulties…

[LeGrand, Markakis, Mehta ’07]:

- Computing a minimax solution is NP-hard
- Any algorithm that computes a minimax solution is manipulable

⇒ Resort to approximation algorithms

Some difficulties…

[LeGrand, Markakis, Mehta ’07]: dictatorial 3-approximation strategyproof algorithm

- Non-dictatorial algorithms?
- ∃ better than 3-approximation algorithms?
- ∃ better than 3-approximation strategyproof algorithms?
- Group-strategyproof algorithms?

Approximation algorithms

- P_i: approval set of voter i
- $d(K, P_i)$: Hamming distance of voter i from an outcome K

- Definition: An outcome K is Pareto-efficient if there is no other outcome K' with
  - $d(K', P_i) \leq d(K, P_i)$ $\forall$ voter i
  - $d(K', P_i) < d(K, P_i)$ for some voter i

- Theorem: Any Pareto efficient algorithm has an approximation ratio of at most 3 - 2/(k+1)

- Corollary: The minisum solution is a non-dictatorial (3 - 2/(k+1))-approximation.

A Linear Programming approach

Integer Program:

min $\quad q$

s.t. $\quad d(x, P_i) \leq q \qquad \forall$ voter i

$\sum x_j = k$

$x_j \in \{0,1\}$

where $d(x, P_i) = \sum_{j \notin P_i} x_j + \sum_{j \in P_i} (1 - x_j)$

A Linear Programming approach

Algorithm:

- Relax to $x_j \in [0,1]$
- Solve the LP
- Pick the committee that corresponds to the k highest values x_j

Theorem:

(i) The approximation ratio of the LP-based algorithm is 2

(ii) The integrality gap of the integer program is 2

Outline

- Approval Voting
  - The minisum and the minimax solution
- Approximation algorithms for the minimax solution
  - Pareto-efficiency and approximability
  - An LP-based algorithm
- Immunity to manipulation
  - Strategy-proof and group-strategyproof algorithms and lower bounds

Mechanism Design (without money)

3 notions of immunity to manipulation

- Strategyproof (SP): no voter can benefit by unilaterally changing his vote
- Group-strategyproof (GSP): no set of voters can **all** benefit by changing their votes
- Strongly group-strategyproof (strongly GSP): no set of voters can change their votes so that at least one of them benefits and no-one is worse off

SP algorithms

- Upper bound: $3 - 2/(k+1)$
  - The minisum algorithm with proper tie breaking is strategyproof

- Theorem: No strategyproof algorithm can have a ratio better than 2 for $k=1$, and better than $2 - 2/(k+1)$, for $k \geq 2$.

SP algorithms

- Proof of lower bound for *k=1*

• Suppose outcome is a_1 on 1st profile (same for other cases)

• On 2nd profile outcome must also be a_1 by SP property

• Then $d(a_1, P_2) = 4$

• OPT = 2 (choose a_4)

	a_1	a_2	a_3	a_4	a_5	a_6
1	1	1	1	0	0	0
2	0	0	0	1	1	1

	a_1	a_2	a_3	a_4	a_5	a_6
1	1	0	0	0	0	0
2	0	0	0	1	1	1

GSP algorithms

- Minisum is not GSP
- The dictatorial algorithm of [LeGrand, Markakis, Mehta ’07] with appropriate tie breaking is GSP
- Can also be made Pareto-efficient $\Rightarrow$ $(3 – 2/(k+1))$-approximation
- Known lower bound: same as for SP

Strongly GSP algorithms

Theorem: If a mechanism is strongly GSP, its ratio is either 3 – 2/(k+1) or ∞

Follows by:

- Lemma 1: A strongly GSP algorithm with finite ratio is Pareto-efficient
  - If not, ∃ better outcome K'
  - Suppose everyone changes vote to K'
  - Finite ratio ⇒ new output is K'
- Lemma 2: Any strongly GSP algorithm has ratio at least 3 – 2/(k+1)

Conclusions

	Lower Bound	Upper Bound
Approx. ratio	NP-hard	2
SP	$2 - 2/(k+1)$	$3 - 2/(k+1)$
GSP	$2 - 2/(k+1)$	$3 - 2/(k+1)$
Strongly GSP	$3 - 2/(k+1)$	∞

Future work

- What is the best approximation ratio achievable in polynomial time? ∃ PTAS?
  - ∃ PTAS for unrestricted version (no constraints on size of committee)

- Characterization of (group) strategyproof algorithms

- Investigate weighted version of minimax [Brams, Kilgour & Sanver, '07]

- Other concepts in Approval Voting
  - [Brams, Kilgour '10]: Satisfaction Approval Voting (SAV): Pick committee that maximizes the sum of satisfaction scores of the voters

Thank you!