Lecture 1

Thursday, 20 August 2026 3:00 PM

What is "social choice"?

- algorithms / mechanisms for groups of people to make decisions, collectively.

Example: voting,
(incl. committees, how to allocate budgets, selecting students, etc)
seat allocation, in colleges & courses,
more generally, allocating scarce resources
eg. food kits, vaccines,
allocating goods / resources / bads fairly:
eg budget among various projects,
inheritance among relatives,
fishing in intl. waters,
chores between a couple living together, etc.

- usually money cannot be exchanged (so different from auctions)
(nevertheless, will show money is still impt.)

What is "computational social choice"?

- associated with computation
- but more generally, more concerned with quantitative aspects.
- so less work on new models, empirical work.

Logistics:

Course will roughly be in 3.5 parts:
1. Fair division
2. Voting
3. Matchings

3.5. Some new stuff (fairness in ML / projects / paper presentations)

In between may take some basics, eg. equilibria, linear / convex programming, etc.
If there are topics you'd like me to cover, please let me know.

Grading:
1. 3-4 HWs + quizzes ~ 30%
2. 2 in-class exams ~ 30%
3. Project / paper ~ 40%

Pacing: will be erratic, will take extra classes

PART 1: FAIR DIVISION

How should we divide items among a set of interested agents?

Example 1: 2 siblings
inherit bank deposit, stocks, gold jewelry, a house, a car

Example 2: 3 house mates
share a house w/ 3 unequal rooms, 40,000 Rs. rent p.m.

Example 3: A couple
chores: cleaning, cooking, groceries, watering plants

More examples: vaccines, production quotas, fishing rights, ...

Real-world applications:

- Spliddit,
- Kaji bahtan,
- Course match (Wharton),
- Fair Outcomes Inc.

(on slides)

CAKE - CUTTING

The cake $C = [0,1]$, n agents, each agent $i \in N$ has a valuation fn. v_i defined over subsets of C, so that $\forall i$:
1. $v_i(S) \geqslant 0 \quad \forall S \subseteq C$ ("cake")
2. For disjoint S, T, $v_i(S \cup T) = v_i(S) + v_i(T)$ (additivity)
3. For any point x, $v_i(x) = 0$ (non-atomicity)
4. $v_i(C) = 1$ (normalization)

(v_i is thus a non atomic probability measure on $[0,1]$)
Cake = time, land, bandwidth, etc.

Note that this is not wlog.

Claim: $v_i([x,y])$ is continuous (w/o proof)

How do n agents "fairly" decide this cake?
(do some examples?)
An allocation $A = (A_1, A_2, \dots, A_n)$ is a partition of the cake, so that:
1. $\forall i \neq j$, $A_i \cap A_j = \phi$ (*)
2. $\bigcup_{i=1}^{n} A_i = [0,1]$

Fairness Concepts:

1. Proportionality (PROP): $\forall i$, $v_i(A_i) \geqslant \frac{1}{n} v_i(C)$

basic, easy. each agent gets an average amount of cake.

2. Envy-freeness (EF): $\forall i, j$, $v_i(A_i) \geqslant v_i(A_j)$
each agent is happy with their allocation.

Lemma: If Allocation $A = (A_1, \dots, A_n)$ is EF, then it is PROP also.
Proof: Fix $i \in N$. Then $v_i(A_i) \geqslant v_i(A_j) \quad \forall j \in N$

$\Rightarrow n\, v_i(A_i) \geqslant \sum_{j \in N} v_i(A_j) = v_i(C) = 1$ (since A is a partition, & using additivity).

$\Rightarrow v_i(A_i) \geqslant 1/n$ ∎

For cake-cutting, will use the following "Robertson-Webb" (RW) query model:
1. $Eval_i(x,y)$: returns $v_i([x,y])$
2. $Cut_i(x, \alpha)$: returns $\inf\{y \geqslant x : v_i([x,y]) = \alpha\}$, if it exists

Claim: Both above fns. are continuous in both arguments

Q. Does PROP $\Rightarrow$ EF?

Theorem: There always exists a PROP allocation.

Algo: (PROP allocation for n agents)
Let $x_0 = 0$, $N' = N$
for $k = 1 \dots n-1$
  $\forall i \in N'$, let $y_i = Cut_i(x_{k-1}, 1/n)$
  Let $i^* = \arg\min_{i \in N'} y_i$

  $A_{i^*} = [x_{k-1}, y_{i^*}]$
  $N' \leftarrow N' \setminus \{i^*\}$, $x_k \leftarrow y_{i^*}$
Let i^* be the last remaining agent in N', then $A_{i^*} = [x_{n-1}, 1]$

(note that this gives a "connected" allocation, i.e., each agent gets an interval)

Proof. Will show that at the end of the k^{th} iteration, for every agent i remaining in N', $val_i([x_k, 1]) \geqslant (n-k)/n$.
Note that this is true after the 0^{th} iteration, i.e., at the beginning of the first iteration.
Now fix an iteration k, and an agent $\hat{i} \in N'$ at the end. Then note that, in each previous iteration $j \leqslant k$, $val_{\hat{i}}[x_{j-1}, x_j] \leqslant 1/n$. This is because if $val_{\hat{i}}[x_{j-1}, x_j] > 1/n$, then in the j^{th} iteration, a smaller piece would have been cut.
Hence $v_{\hat{i}}[0, x_k] \leqslant k/n$, & hence $v_{\hat{i}}[x_k, 1] \geqslant \frac{n-k}{n}$ ∎

An EF allocation for 2 agents.
(The cut-and-choose protocol).

Algo:
Let $x = Cut_1(0, 1/2)$ (then $v_1([0,x]) = 1/2 = v_1([x,1])$)
If $v_2([0,x]) \geqslant v_2([x,1])$ (agent 1 cuts, agent 2 chooses)
  $A_1 = [x,1]$, $A_2 = [0,x]$
Else
  $A_1 = [0,x]$, $A_2 = [x,1]$

But we can do more:

Defn: Allocation $A = (A_1, \dots, A_n)$ is a <u>perfect partition</u> if $\forall i \in N$, $j \in [n]$, $v_i(A_j) = 1/n$

Theorem: There exists a perfect partition for $n = 2$.
Proof: Define $f_1(y) = Cut_1(y, 1/2)$
Define $w_2(x) = v_2([0,x]) + v_2([f_1(x), 1])$
Since f_1 is a continuous fn., so is w_2.
Now consider the cut $[0, f_1(0)]$.
Then $v_1([0, f_1(0)]) = 1/2 = v_1([f_1(0), 1])$
If $v_2([f_1(0), 1]) = 1/2$, we are done.
So suppose $v_2([f_1(0), 1]) > 1/2$
Now $w_2(0) = v_2([f_1(0), 1]) > 1/2$
& $w_2(f_1(0)) = v_2([0, f_1(0)]) + v_2([f_1(f_1(0)), 1])$
$= v_2([0, f_1(0)]) < 1/2$
Thus, $\exists x^*$ s.t. $w_2(x^*) = 1/2$
Then consider the cut $A_1 = [x^*, f_1(x^*)]$
$A_2 = [0, x^*] \cup [f_1(x^*), 1]$
This is a perfect partition. ∎

Q. $\exists$? a connected perfect partition for 2 agents?