Social Choice and Networks: A Quantitative GS Theorem

Lecturer: Elchanan Mossel
Scribe: Gireeja Ranade

October 26, 2010

The GS theorem tells us that if a function F is *neutral and non-manipulable*, then it must be a dictator function. Our goal in this lecture is to prove a quantitative version of the GS theorem.

Some notation and assumptions:
n voters
k alternatives denoted $\{A, B, \ldots K\}$
Each voter submits a ranking of his preferences
As social planners, we want a function where each voter is incentivized to reveal truthful preferences
$M(F)$ = all configurations of rankings where at least one voter can manipulate the result
$D_k(n)$ = dictators on k alternatives and n voters
$D(F, G) = P(F(\sigma) \neq G(\sigma))$. $D(F, D_k(n)) = \min\{D(F, G) : G \in D_k(n)\}$.

Consider people voting according to a random order uniformly in S_k^n. What is the probability that a non-dictator function might be manipulable? The GS theorem gives us that $(\frac{1}{k!})^n \leq M(F)$ (there is atleast one σ manipulable by at least one voter).

Inspired by Arrow's theorem, we might be interested in a result of the following flavor — If function F is at least ϵ far away from the dictator function, then P(F is manipulable) $> \delta$? The answer is no. Consider the plurality function on k alternatives. Then,

P(σ: F is manipulable at σ) $\leq$ P($\sigma :\ \exists\ i,\ j$ s.t. i, j appear at the top rank for an equal (upto $\pm$ 1, based on tie break rule)number of voters) $= O(\frac{1}{\sqrt{n}}) \to 0$ as $n \to \infty$. Hence, we have $(\frac{1}{k!})^n \leq M(F) \leq O(\frac{1}{\sqrt{n}})$.

A note: As the "social-planner" in this problem, we want (1) all voters to vote truthfully, and (2) we don't want a single voter to be able to manipulate the system. Do we care about this issue for real election systems? Not really — the probabilities of manipulation by a single voter are too small. However, consider a scenario where an aggregator is combining the results from Google, Yahoo, Bing and ... Ask.com(?) to produce a master list of results to a particular query. In this case, we don't want any individual voter (i.e. search engine) to be able to manipulate the outcome of our system. This is really the scenario to have in mind when thinking about these questions.

Theorem (Issakson-Kindler-Mossel 10:) If F is *neutral* and $k \geq 3$, then $M(F) \geq n^{-3}k^{-10}(D(F, D_k(n)))^2$. Moreover, the trivial random algorithm manipulates with probability at least $n^{-3}k^{-10}(D(F, D_k(n)))^2$.

Previous results have shown that manipulation in some situations can be hard (Bartholdi, Orlin 91 and Bartholdi, Tovey, Trick 93). Also, there was a theorem (Friedgut-Kalai-Nisan 08): For $k = 3$ alternatives, and neutral F, it holds that $M(F) \geq n^{-1}D(F_k(n), D)^2$. No computational consequences. They conjectured that random manipulation could provide $M(F) \geq poly(n, k)^{-1}$ and was easy on average.

Proof: The proof of IKM-10 uses four (plus three) ideas. First, we define a graph representation of all the voters and their possible voting preferences. The graph is such that we can construct paths between edges of the graph that must hit manipulation points or pass close to manipulation points. Finally, we get an estimate of the number of manipulation points by comparing the number of paths with the maximum number of paths that may pass through any particular manipulation point.

1. Idea 1: The rankings graph

   Consider a graph with edges and vertices as described here. Each vertex is the set of rankings on k alternatives by all n voters, i.e., the vertex set is $S(A, B, ...K)^n$. We define an edge $e[x, x']$ (on voter i) if the rankings of all voters other than i stay the same at vertices x and x', i.e., $x(j) = x'(j)\ \forall\ j\ \neq\ i$. Fix a voting function F. $e[x, x']$ is a boundary edge if $F(x) \neq F(x')$. We have three types of boundary edges — *monotone*, *monotone-neutral* and *anti-monotone*. Recall that at any edge only one voter, say voter i, changes ranking from x to x'.

   Monotone: $F(x) = A, F(x') = B$. x ranks A above B. x' ranks B above A.
   Monotone-neutral: $F(x) = A, F(x') = C$. x and x' have the same order for A, C.
   Anti-monotone: $F(x) = B, F(x') = C$. x ranks C above B. x' ranks B above C.

   Monotone-neutral and anti-monotone edges are manipulable. To see this, consider the monotone-neutral setup and say voter i prefers A over C. Then he would want to vote according to x, even if x' might represent his true preferences. Anti-monotone edges present a similar situation.

2. Idea 2: Isoperimetry

   Assume we are only looking at four alternatives A, B, C, D.

   Isoperimetric Lemma: If F is ϵ far from all dictators and *neutral*, then there exist voters $i \neq j$ such that $P(e \in \partial_i(A, B)) \geq \epsilon(6n)^{-2}, P(e \in \partial_j(C, D)) \geq (6n)^{-2}$.

   Proof sketch: If F is far from being a dictator, then there are at least two voters who can impact the outcome. (If F we a dictator, only one voter would be able to affect the outcome.) Voter i changes the outcome with $P(\epsilon n^{-2})$. Voter j changes the outcome with $P(\epsilon n^{-2})$.

3. Idea 3: Paths and Flows on $\partial_i(A, B)$

   Focus on a space of edges — say the boundary between the outcomes A and B. Note that an element in this new space is an edge on the old graph. This is not a connected space, but we will define walks on this space. This walks will be such that whenever the walks runs out of the space it hits a manipulation point. Any time we get stuck, we will be able to show that we are close to a manipulation point.

   Walk rule: We can take a step from $e[x, x']$ (on voter i) to $e[y, y']$ (on voter i) if y differs from x only in one voter, say j, and maintains the relative ranking for A and B at this voter. The same rule must hold for x' and y' as well.

   Lemma: *If we reach a point along our walk where we can't go any further, we must be close to a manipulation point.* Let $[x, x'] \in \partial_i(A, B), j \in n\backslash\{i\}, y_{-j} = x_{-j}, y'_{-j} = x'_{-j}$. y_j, y'_j have the same A, B order as x_j, x'_j. Then, either $[y', y] \in \partial_i(A, B)$ or $\exists$ a manipulation point identical to x except in at most three voters (and thus, at most three steps away from x).

   Proof idea: Let $[y', y] \notin \partial_i(A, B)$. If $F(y) \notin \{A, B\}$ then at this point F takes at least 3 values. We can apply the GS theorem by fixing all voters except i, j, and conclude there exists a manipulation. Alternatively, if $F(y) \in \{A, B\}$, then we can say without loss of generality that $F(x) = F(y) = F(y') = A, F(y') = B$. Then $[x', y']$ is a manipulation edge.

4. Idea 4: Canonical paths

   We *start* at an edge $e[x,x'] \in \partial_i(A,B)$, and choose as *target* an edge $e'[y,y'] \in \partial_j(C,D)$ and define a *canonical path*, $\Gamma\{e,e'\}$ between these edges along the 'meta graph'. The path begins at e and ends at e'. The path stays in $\partial_i(A,B) \cup \partial_j(C,D)$ or encounters a manipulation.

   The example shows the evolution of one of the vertices (say x of $e[x,x']$) in a sample canonical path walk. We change the vertices one voter at a time while maintaining the A,B order for each voter, until we hit a point (the fourth step in the example, $\star$), where we cannot proceed without changing the A,B order. Similarly, we meet this point from the other direction.

   $$\text{START} = \begin{bmatrix} A & C & B \\ B & D & A \\ C & A & D \\ D & B & C \end{bmatrix} \to \begin{bmatrix} D & C & B \\ A & D & A \\ C & A & D \\ B & B & C \end{bmatrix} \to \begin{bmatrix} D & C & B \\ A & A & A \\ C & D & D \\ B & B & C \end{bmatrix} \to \begin{bmatrix} D & C & B \\ A & A & D \\ C & D & A \\ B & B & C \end{bmatrix} \overset{\star}{\to} \begin{bmatrix} D & C & B \\ B & A & D \\ C & D & A \\ A & B & C \end{bmatrix} \leftarrow \begin{bmatrix} D & C & B \\ B & A & D \\ C & D & A \\ A & B & C \end{bmatrix} = \text{TARGET}$$

   (The last two matrices in the example *are* equal, that isn't a typo. It just so happens that in this example the path in the reverse direction is of length one.)

   Now, if this path does not encounter a manipulation along the way, then at the transition point from $\partial_i(A,B)$ to $\partial_j(C,D)$, F takes at least three values, so the GS theorem implies there must be a manipulation.

5. Counting the manipulation points - Are there enough?

   Okay, great, so now we know how to hit manipulation points, or at least how to get close to them. The next question is - are there enough such points? Or will all paths of the form described above pass through the same manipulation points?

   Let $R = \max_m \#\{\{e,e'\} : m \text{ is a manipulation for } \Gamma\{e,e'\}\}$.

   Then, since $|M(F)| \geq R^{-1}|\partial_i[A,B] \times \partial_j[C,D]|$, we have that
   $P[M(F)] \geq (4!)^n R^{-1} P[\partial_i[A,B]] \times P[\partial_j[C,D]]$.

   Now, given m, a point in a path on the 'meta-graph' — how many paths $\Gamma\{e,e'\}$ could pass through it? To prove our result, we want to "decode" $\leq \text{poly}(4!)^n$ pairs (e,e') such that $\Gamma\{e,e'\}$ passes through it. If $e=[x,x']$ and $e'=[y,y']$, then it is enough to decode (x,y) from the point m, and pay $k!^2$ for the ambiguity in (x',y'). For a coordianate s, given m_s we know either x_s or y_s or we have $\frac{4!}{2}$ options for x_s and 2 options for y_s. So we only have $\text{poly}(n,k)$ paths passing through each point m.

   So $P[M(F)] = (4!)^n R^{-1} P[\partial_i[A,B]] \times P[\partial_j[C,D]]$ gives $P[M(F)] \geq \epsilon^2(6n)^{-5}$.

6. A bit of cheating...

   In fact, we were cheating a bit in the previous analysis. We can only prove that along the path we may hit a manipulation point or be close to (at most 3 steps away from) a manipulation point. So we have $R \leq n4^n k!^3$, which gives us $P[M(F)] \geq k!^3\epsilon^2(6n)^{-5}$. This is fine for a constant number of alternatives k, but doesn't work for large k.

7. Idea 5: Geometries on the ranking cubes

   We define a new graph with the same vertices as before. But now (x,x') is an edge if x,x' differ in a single voter and an adjacent transposition. Further analysis with this graph can get us to the desired polynomial factors.