CS-698A: Selected Areas of Mechanism Design **Jan-Apr 2018**

Lecture 10: Limitations of core and the Nucleolus

Lecturer: Swaprava Nath *Scribe(s): Shreyash Tiwari*

Disclaimer: *These notes aggregate content from several texts and have not been subjected to the usual scrutiny deserved by formal publications. If you find errors, please bring to the notice of the Instructor.*

In the previous lecture we discussed Shapley value, and we proved that it may not lie in the core. For such game settings we can expect either;

- A stable sub-coalition may form.
- A weaker notion of core may be proposed.

10.1 Refinements of the core

10.1.1 ϵ- core

Similar in spirit to the idea of ϵ-Nash equilibrium.

Definition 10.1 *A payoff vector x is in the ϵ-core of a coalitional game (N,v) if*

$$\sum_{i \in S} x_i = v(S) - \epsilon$$

Interpretation: There is a cost to move from the grand coalition, which is denoted by ϵ. If the value is not decreasing more than that cost, it is possibly not meaningful to deviate from the grand coalition.

Mathematically, there is no reason why $\epsilon \geq 0$. If $\epsilon < 0$ the condition of ϵ-core is giving some "bonus" for forming a coalition. The allocation not only is coalitionally rational, it awards something that is strictly better by a constant margin. The ϵ-core allocation is more stable than a core allocation. But for a given ϵ, the ϵ-core may still be empty. We can confine relaxing the ϵ-core until some non empty ϵ-core is found.

10.1.2 Least core

Definition 10.2 *A payoff vector x is in the least core of (N, v) if x is a solution to the following linear program*

$$min\ \epsilon$$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon$$

for all $S \subset N$

$$x(N) = v(N)$$

Properties

- objective is non positive iff the core of the game is non-empty.
- For sufficiently large ϵ, the constraints can be satisfied.
- When the core is non-empty, least core does not contain all core allocations. It rather gives the least opportunity for every coalition to deviate in a core refinement.
- Similar to core, Least core has a set valued solution.

If the inequality solves with equality than the solution is **tight**. The solution of this LP may return a vector x and ϵ such that not all inequalities are tight.

Note: Least core is a refinement of core. **Is it unique?**

Exercise: Construct examples of multiple solutions of least core.

Strengthening the least core

One simple idea is to repeat the above process. Make the slack inequalities tight. Formally, if ϵ_1 be the optimal value of the LP-1, we optimize.

Also let S_1 be the set of coalitions for which the inequalities are tight

$$min\ \epsilon$$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon_1$$

for all $S \in S_1$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon$$

for all $S \in 2^N \backslash S_1$

$$x(N) = v(N)$$

This makes few more inequalities tight. There could still be some slack inequalities. We sequentially repeat this procedure until all inequalities become tight. Since there are finite number of inequalities, this always converges to a unique payoff vector known as **nucleolus**.

10.2 Nucleolus

Definition 10.3 *An allocation x is in the nucleolus of a coalitional game (N, v) if it is the solution of a series of LPs*

[LP-1]

$$min\ \epsilon$$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon$$

for all $S \subset N$

$$x(N) = v(N)$$

[LP-2]

$$min\ \epsilon$$
$$st.\ \sum_{i \in S} x_i = v(S) - \epsilon_1$$

for all $S \in S_1$

$$\sum_{i \in S} x_i = v(S) - \epsilon$$

for all $S \in 2^N \backslash S_1$

$$x(N) = v(N)$$

[LP-k]

$$min\ \epsilon$$
$$st. \sum_{i \in S} x_i = v(S) - \epsilon_1$$

for all $S \in S_1$

$$.$$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon_1$$

for all $S_{k-1} \in S_{k-2}$

$$st. \sum_{i \in S} x_i = v(S) - \epsilon$$

for all $S \in 2^N \backslash S_{k-1}$

$$x(N) = v(N)$$

This needs at most n iterations.

Exercise: Argue over the dimensions of the variable space.

Theorem 10.4 *For any game* (N, v)*, nucleolus exists and is unique.*

Proof:
Existence: The series of LPs cant be solved and reach some assignment of x's s.t. all inequalities are met with equality. In every round, at least one inequality will become tight, eventually it would converge (property of LP) and thus a solution always exists.

Uniqueness: Earlier LPs influence the latter LPs only via the values. Therefore, the set of LPs will always lead to the same set of solutions $(\epsilon_1, \epsilon_2, \ldots)$. (In particular $\epsilon_1 \geq \epsilon_2 \geq \cdots$).
After all the iterations are over, we are left with 2^n equations over n variables. It has rank of n - if a solution exists, it must be unique.

$$\begin{bmatrix} & & Individuals & & \\ & x_{11} & x_{12} & \ldots & x_{1n} \\ Coalitions & \vdots & \vdots & \ddots & \vdots \\ & x_{d1} & x_{d2} & \ldots & x_{dn} \end{bmatrix} = \begin{bmatrix} v(S) - \epsilon_1 \\ v(S) - \epsilon_2 \\ \vdots \\ v(S) - \epsilon_k \end{bmatrix}$$

Columns are linearly independent. They are the allocation given to each individual. They are unique so nucleolus is unique.
We have made the RHS live in the space spanned by the columns of association matrix. Therefore, there exists a unique linear combination of these columns to yield the RHS (fact from linear algebra). ■

10.2.1 An alternate definition

Nucleolus is also defined w.r.t. excesses.

Definition 10.5 *The excess of a coalition S in (N, v) and a payoff vector x is denoted as*

$$e(S, x, v) = v(S) - \sum_{i \in S} x_i$$

Observation: If core is non-empty, then $\exists x$ s.t. all excesses are non-positive. Given a coalitional game (N, v) and a payoff vector x compute all excesses except coalitions N and ϕ .

This 2^n - 2 dimensional vector is the **raw excess** vector. The vector is sorted in decreasing order - sorted excess vector denoted $\theta(x, v)$. Given two payoff vectors x and y , we say excesses due to x are **lexicographically smaller** than those due to y, written $x \preceq y$ if for the smallest index where $\theta(x, v)$ and $\theta(y, v)$ differ, $\theta(x, v) \leq \theta(x, v)$.
This is a valid binary relation, which is reflexive, transitive, complete, but not symmetric.

Definition 10.6 *Given a coalitional game (N, v), the nucleolus in the payoff vector x such that for all other payoff vectors y, $y \succeq x$, i.e. x lexicographic-ally minimizes the excesses of all coalitions.*

10.3 Compact representation of coalitional games

- We have seen several solution concepts and how to compute them in practice.
- The representation of the game is important in answering that question. A straightforward representation will take enormous space.
- Gives a feeling that even brute force method are also **good**. Hence a compact representation is important.

10.3.1 Weighted graph games

This game is defined by an undirected graph with edge weights and the value of a collection of nodes is the sum of edge weights that run between the nodes. Cities(Nodes) that are connected via high-speed highways(edges) with toll(weights): to share the revenue among them

Important: It is easy to represent the values, just $\binom{n}{2}$ numbers for the edge weights.

Definition 10.7 *(WCG) Let (V, W) denote an undirected weighted graph, V = set of matrices, $W \in R^{|V| \times |V|}$ is the set of edge weights, W symmetric weight between i and j denoted by $W(i, j)$. The coalitional game WCG is a game (N, V) s.t.*

1. N = V
2. $v(S) = \sum_{i,j \in S} w(i, j) \forall S \subset N$

Proposition 10.8 *If all weight are non-negative, WCG is convex.*

$$v(S) + v(T) \leq v(S \cup T) + v(S \cap T)$$

10.3.2 Shapley Value

Theorem 10.9 *The Shapley Value of the game induced by a WCG (V, W) is*

$$Sh_i(N, v) = \frac{1}{2} \sum_{j \neq i} w(i, j)$$

Proof:

$$Sh_i(N, v) = \frac{1}{n!} \sum_{\pi \in \Pi(N)} w_i^\pi$$

$$w_i^\pi = w_i^\pi(P_i(\pi \cup \{i\}) - v(P_i(\pi))$$

$$v(P_i(\pi)) = \sum_{j,k \in P_i(\pi)} w(j, k)$$

$$v(P_i(\pi) \cup \{i\}) = \sum_{j,k \in P_i(\pi)} w(j, k) + \sum_{k \in P_i(\pi)} w(i, k)$$

$$w_i^\pi = \sum_{k \in P_i(\pi)} w(i, k)$$

Sum over all possible permutations, how many times should a specific k appear before i ? $\frac{n!}{2}$

$$w_i^\pi = \frac{n!}{2} \sum_{j \neq i} w(i, j)$$

$$Sh_i(N, v) = \frac{1}{2} \sum_{\pi \in \Pi} w(i, j)$$

■

Observation: We can compute Shapley value in $O(n^2)$ time. Answering questions regarding the core of WCG is more complex.

Cut is a set of edges that divide the nodes of a graph into two parts $(S, V \backslash S)$. The weight of a cut is the sum of its weights.

$$= \sum_{i \in S, j \in N \backslash S} w_{ij}$$

Theorem 10.10 *The shapley value is in the core of a WCG iff there is no negative cut in the weighted graph.*

Proof: From the previous result , if $\exists$ a cut $S, N \backslash S$ such that the weight is negative then

$$\sum_{i \in S} Sh_i = \frac{1}{2} \sum_{i \in S} \sum_{j \neq i} w(i, j)$$

$$= \frac{1}{2} \sum_{i \in S} [\sum_{j \neq i, j \in S} w(i, j) + \sum_{j \in N \backslash S} w_(i, j)]$$

$$= \frac{1}{2}.2 \sum_{(i,j)\in S^2} w_{ij} + \frac{1}{2} \sum_{i\in S, j\in N\setminus S} w(i,j)$$

$= v(S) + \frac{1}{2}$.weight of cut
$<$ v(S) iff weight of cut is negative ■

Theorem 10.11 *Testing non-emptiness of core for general WCG is NP-complete*

Proof is not discussed in this lecture.