Some open problems in Game Theory

Symposium on Combinatorics and Probability

Souvik Roy

Applied Statistics Unit
Indian Statistical Institute, Kolkata

Nim and Graph Nim games

- Nim is a mathematical game of strategy which is played with several heaps of objects, and two players alternate taking one or more objects from a single heap.
- The game ends when all the objects are removed and the player who makes the last move wins.
- Nim has been solved for any number of initial heaps and objects, meaning there is a winning strategy for the first player provided the game meets one initial condition.

Example Game of Nim.PNG

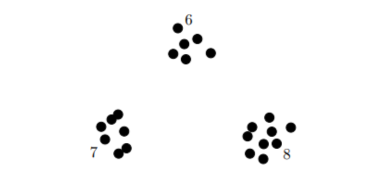

Figure 1: An Example Game of Nim

- Let N be the total no. of heaps, $a_i > 0$ be the initial no of objects in i^{th} heap and $\oplus$ be the XOR sum operator. Then the game is in losing position iff $a_1 \oplus a_2 \oplus \cdots \oplus a_N = 0$ (call it the Nim-sum)
- Let Nim-sum be 0. Suppose that Player 1 removes objects from i^{th} heap, only a_i will change to $a_i'(\neq a_i)$ which implies $a_1 \oplus a_2 \oplus \cdots \oplus a_i \oplus \cdots \oplus a_N \neq 0$. Now, player 2 can always make a move s.t. Nim-sum becomes 0 and the process repeats until $a_i = 0 \; \forall i$ (Note that $0 \oplus \cdots \oplus 0 = 0$ i.e., when all heaps are empty, Nim-sum is 0)

Game of Nim in a Losing Position.PNG

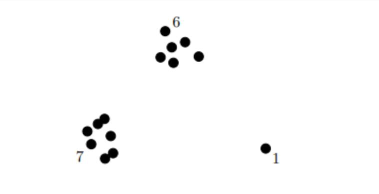

Figure 2: A Game of Nim in a Losing Position

- Graph nim is a generalization of nim where the idea is to associate some of the heaps. Then, instead of selecting a single heap to remove objects from, a player can now select heaps that are associated and remove any number of tokens from any of those heaps.
- Graph Nim consists of a graph where in each turn, a player must select one of the vertices and then remove weights from any number of edges incident to that vertex

weighted graph on 4 vertices.PNG

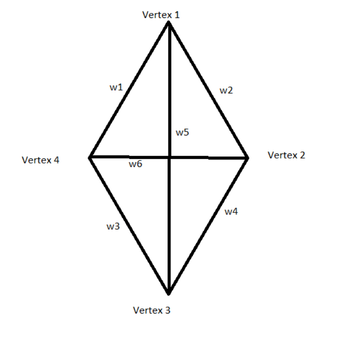

Figure 3: Complete weighted graph on 4 vertices

- Again the game ends when all the weights are made 0 and the player who makes the last move wins.
- Now, interestingly Graph Nim on path graphs P_n of length ($=$ no of edges) $n \in \mathbb{N}$ with equal weights on all edges is always winning.
- Also, Graph Nim on C_n with equal weights is winning if n is even and losing if n is odd

graph with equal weights.PNG

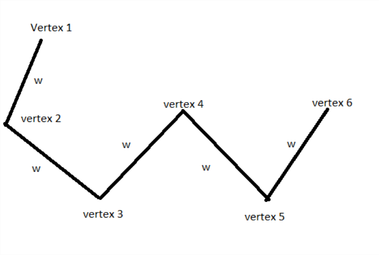

Figure 4: path graph with equal weights

- In fact in general setups, Graph Nim on any graphs on $v \leq 4$ vertices has been completely analysed into winning and losing positions.
- It has been found that Graph Nim on P_n of length ≤ 5 is always winning for any configuration of weights.
- Analysing more complex positions and other variants of graph nim is an active field of research in combinatorial game theory.

- Fix $p, q \in [0, 1]$ with $p + q \leqslant 1$. Each site of the lattice $\mathbb{Z}^2$ is either a *trap* with probability p, a *target* with probability q, or simply *open* with probability $1 - p - q$, independently of all other sites.
- Two players (let's call them P1 and P2) take turns to move a token, which begins at some *initial vertex* (for example, the origin $(0, 0)$).
- Suppose the current position of the token is (i, j). The player whose turn it is to move can now move the token to one of the available vertices defined by a function $A(i, j)$. In general, $A(i, j)$ may depend on the player who is making the move at (i, j).
- In our paper, we consider $A(i, j)$ to be the set of vertices whose graph distance from (i, j) is at most two, that is,

$$A(i, j) = (i + 1, j), (i, j + 1), (i + 2, j), (i + 1, j + 1), (i, j + 2).$$

- If a player is forced to move the token to a trap, they lose immediately. If a player is able to move the token to a target, they win immediately. Otherwise the game goes on.

Existing results

- Holroyd, Markovici and Martin (henceforth HMM) consider a simpler game where the available moves from (i, j) are $(i+1, j)$ and $(i, j+1)$.
- They show if $p > 0$ or $q > 0$, then the PCA $A_{p,q}$ is ergodic and the draw probability of the percolation game is 0.

Connection between percolation games and probabilistic cellular automata

- The PCA, denoted $A_{p,q}$, studied in HMM involves:
  - the alphabet $\{0, 1\}$,
  - the universe $\mathbb{Z}$,
  - so that at any point of time, we are considering a configuration in the space $\Omega = \{0, 1\}^{\mathbb{Z}}$.
- Let η_t denote the configuration at time t (this is discrete time), where $\eta_t(n)$ denotes the state of the site n at time t.

Connection between percolation games and probabilistic cellular automata

- Given η_t, the configuration η_{t+1} is obtained by updating the state of each site in $\mathbb{Z}$ independently, according to the following rule:
    - if $\eta_t(n-1) = \eta_t(n) = 0$, then we set $\eta_{t+1}(n)$ to be 0 with probability p and 1 with probability $1-p$;
    - otherwise, i.e. if at least one of $\eta_t(n)$ and $\eta_t(n-1)$ equals 1, we set $\eta_{t+1}(n)$ to be 0 with probability $1-q$ and 1 with probability q.

Connection between percolation games and probabilistic cellular automata

- If $\sigma \in \Omega$ is a random configuration with given probability distribution μ, then $A_{p,q}\mu$ is the distribution of the (random) configuration we obtain by applying $A_{p,q}$ to σ, i.e. by updating σ via the rules of $A_{p,q}$ for one time-step.
- We call μ an invariant or stationary distribution for $A_{p,q}$ if $A_{p,q}\mu = \mu$. More generally, μ is said to be k-periodic for $A_{p,q}$ for some $k \in \mathbb{N}$ if $A_{p,q}^k\mu = \mu$, where $A_{p,q}^k$ implies that given a random configuration σ whose distribution is μ, we update σ according to the rules of $A_{p,q}$ for k time steps, and the resulting random configuration has distribution $A_{p,q}^k\mu$.
- We say that μ is periodic for $A_{p,q}$ if it is k-periodic for $A_{p,q}$ for some $k \in \mathbb{N}$.

Connection between percolation games and probabilistic cellular automata

Definition

A PCA $A_{p,q}$ is ergodic if

- it has a unique stationary distribution $\mu_{p,q}$,
- given any probability distribution μ over Ω, the distributions $A_{p,q}^k \mu$ converge to $\mu_{p,q}$ as $k \to \infty$.

Our model induces a more generalized, 3-step PCA

Consider the PCA $A_{p,q}$ as follows:

- it has alphabet $\{0, 1\}$,
- its configurations come from the state space $\Omega = \{0, 1\}^{\mathbb{Z}}$,
- given a configuration $\eta_t = (\eta_t(n) : n \in \mathbb{Z})$ at time t, we update the state $\eta_{t+1}(n)$ at each site n for time $t + 1$, independently of all other sites, as follows:
  - if $\eta_t(n) = \eta_t(n-1) = \eta_t(n-2) = 0$, then we set $\eta_{t+1}(n) = 0$ with probability p and $\eta_{t+1}n) = 1$ with probability $1 - p$;
  - otherwise, we set $\eta_{t+1}(n) = 0$ with probability $1 - q$ and $\eta_{t+1}(n) = 1$ with probability q.

Objective: To understand the ergodicity of this PCA.

• In another project, we are working on edge percolation games, which appears to be even “harder”.

A problem on cycle monotonicity

- $A := \{a_1, \ldots, a_m\}$ is the set of m choices/objects/outcomes.
- $\mathcal{D} \subseteq \mathbb{R}^m$ is a domain.
- $f : \mathcal{D} \to A$ is a decision rule.
- A decision rule f satisfies k-cycle monotonicity (k-CM) if $\forall t_1, t_2, \ldots, t_k \in \mathcal{D}$,

$$\sum_{j=1}^{k} t_j(f(t_j)) \geq \sum_{j=1}^{k} t_j(f(t_{j+1}))$$

where $t_{k+1} = t_1$.

A problem on cycle monotonicity

Definition

A decision rule f is implementable if there exists a payment function $p : \mathcal{D} \to \mathbb{R}$ such that $\forall t_i, t_i' \in \mathcal{D}$,

$$t_i\left(f(t_i)\right) - p(t_i) \geq t_i\left(f(t_i')\right) - p(t_i').$$

Question: What are all implementable decision rules?

Theorem

A decision rule is implementable if and only if it satisfies k-CM for all $k \geq 2$.

A problem on cycle monotonicity

- Checking k-CM for a given decision rule is hard–can we relax this requirement further? More precisely, what is the maximum length of cycles one needs to check to ensure k-CM for all $k \geq 2$?
- In particular, is it possible that 2-CM (or 3-CM or so) implies k-CM for all $k \geq 2$?
  - It depends on $\mathcal{D}$.
- If $\mathcal{D}$ is convex, then 2-CM implies k-CM for all $k \geq 2$.
- There are (non-convex) domains $\mathcal{D}$ on which 2-CM and 3-CM together imply k-CM for all $k \geq 2$.
- An important problem in game theory is to characterize all domains on which 2-CM implies k-CM for all $k \geq 2$.

On local-global equivalent domains

- A is the finite set of objects/choices/outcomes.
- $\mathcal{P}$: set of all strict preferences/orderings on A.
- $\mathcal{D} \subseteq \mathcal{P}$ is a domain. Typical elements are denoted by $\prec, \prec', \prec^1, \prec^2, \ldots$ etc.

On local-global equivalent domains

- Let $G = \langle \mathcal{D}, \mathcal{E} \rangle$ be an undirected graph where $(\prec, \prec') \in \mathcal{E}$ if and only if $\prec$ and $\prec'$ differ only in the ranking of two consecutive objects.
- For example:

$\prec^1$	$\prec^2$	$\prec^3$	$\prec^4$	$\prec^5$
a	b	b	b	a
b	a	a	a	b
c	c	c	e	e
d	d	e	c	c
e	e	d	d	d

- A choice function is a map $f : \mathcal{D} \to A$.

On local-global equivalent domains

- f is *strategy-proof on* $(\prec, \prec')$ if $f(\prec') \preceq f(\prec)$.
- f is *locally strategy-proof* if it is strategy-proof on each $(\prec, \prec') \in \mathcal{E}$.
- f is *strategy-proof* if it is strategy-proof on each $(\prec, \prec') \in \mathcal{D} \times \mathcal{D}$.

On local-global equivalent domains

- A domain is called *Local-Global-Equivalence(LGE) domain if every locally strategy-proof choice function on it is strategy-proof.*
- *Question: When is a domain an LGE domain?*
- *Is $\mathcal{P}$ an LGE domain?*
- *We have provided a characterization of LGE domains in* ***Kumar et al. (Theoretical Economics, 2021)****.*
- *This problem is open for random choice functions.*

Extreme point characterization problem

- $N := \{1, \ldots, n\}, n \geq 2$ is the set of players.
- $A := \{a_1, a_2, \ldots, a_n\}$ is the set of objects.
- $\prec_i$: a *strict* preference/ordering on A of player i.
- $\mathcal{D}$: A set of (admissible) preferences.
- $\prec_N := (\prec_1, \ldots, \prec_n) \in \mathcal{D}^N$ is a preference profile of all players.

Extreme point characterization problem

- A random matching is a $n \times n$ bi-stochastic matrix $M = (m_{ij})_{i,j=1}^{n}$ where the rows represent the players and the columns represent the objects. A deterministic matching is a deterministic bi-stochastic matrix $M = (m_{ij})_{i,j=1}^{n}$, that is, $m_{ij} \in \{0, 1\}$ for all $i, j \in \{1, \ldots, n\}$.

- For a matching M, by M_i we denote the i-th row of M.

- $\mathcal{M}$ is the set of all $n \times n$ bi-stochastic matrices.
- $\mu : \mathcal{D}^n \to \mathcal{M}$ is a matching function.

Extreme point characterization problem

- $r_k(\prec)$: k-th ranked object in preference $\prec$.
- Let p, q be two probability distributions on A and let $\prec$ be a preference on A.
  p first order stochastically dominates q at $\prec$ if

$$\sum_{k=1}^{l} p(r_k(\prec)) \geq \sum_{k=1}^{l} q(r_k(\prec), \ l = 1, \ldots, n.$$

Definition

A matching function μ is strategy-proof if for all $i \in N$, all $\prec_N \in \mathcal{D}^n$, and all $\prec'_i \in \mathcal{D}$, $\mu_i(\prec_i, \prec_{-i})$ first-order stochastically dominates $\mu_i(\prec'_i, \prec_{-i})$ according to $\prec_i$.

Definition

A matching function μ is efficient if for all $\prec_N \in \mathcal{D}^n$ and all $M \in \mathcal{M}$ with $\mu(\prec_N) \neq M$ there exists an agent $i \in N$ such that $\mu_i(\prec_N)$ *strictly* first order stochastically dominates M_i.

Extreme point characterization problem

Question:

- Is every efficient and strategy-proof random matching function a convex combination of efficient and strategy-proof deterministic matching function?

Thank You