NashProp

- A heuristic for computing NE in graphical games with arbitrary graphs.

- A generalization of TreeNash.

Basic idea

- Each player neighborhood looks **locally** like a tree.

- So, apply same "table" computation (message-passing operation) as in TreeNash (the algorithm for trees).

- But, there is no **global** root (in arbitrary graphs):
  - From the player's local perspective, "the player doesn't know where the root is" [actually, there might not be any!]

- So, each player "sends table message" to each neighbor as if "root" could be reached through that neighbor.
  - [process can be done distributively and asynchronously]

- Two phases: table-passing phase, assignment-passing phase.
Recall: Tree Nash

1. Only propagate tables "down" the tree to root; then only propagate solutions "up" the tree to leaves

2. No "initialization" needed; use "leaves" to initialize

Nash Prop

1. Propagate tables in rounds; each player sends a table to every neighbor.

   Table-Propagation "rule" is the same as in Tree Nash, except that it is executed for each neighbor independently.

   At each round $t$,
   \[ V(W,V) \in E, \forall (w,v) \in [0,1]^2 \]

   edge set of graph:

   \[ T^t(w,v) = 1 \text{ iff } \exists \text{ a witness } \bar{u} \in [0,1]^{k-2} \text{ s.t.} \]

   \[ 1. \ T^{t-1}(v,u_i) = 1, \forall i = 1, \ldots, k-2 \]

   \[ 2. \ V = \bar{u} \text{ is a best response} \text{ to } W = w, \bar{u} = \bar{u} \]

2. How do we start? (i.e. What about $t=1$?)

   Initialize to full tables
   \[ \forall e; \forall (w,v) \in E, \forall (w,v) \in [0,1]^2 \]

   \[ T^0(w,v) = 1 \]

   Intuition: Lacking any initial knowledge, a player "believes" any strategy is a best response to any other strategy.
Analysis of Abstract table-passing phase

Remarks:

* \( \forall (W,V) \in \mathcal{E}, \forall (\omega, \nu) \in [0,1]^2, \)
  \[ T^{-1}(\omega, \nu) = 0 \Rightarrow T^t(\omega, \nu) = 0 \]

\[ \{ (\omega, v) \in [0,1]^2 : T^{-1}(\omega, v) = 1 \} \supseteq \{ (\omega, v) \in [0,1]^2 : T^t(\omega, v) = 1 \} \]

* So, "tables" converge! \( \forall (W,V) \in \mathcal{E}, \lim_{t \to \infty} \{ (\omega, v) \in [0,1]^2 : T^t(\omega, v) = 1 \} \text{ exists!} \)

**Defn.** Given a graphical game, we say a set of tables \( \{ T_{WV} : [0,1]^2 \to [0,1] \}, \forall (W,V) \in \mathcal{E} \) for the game is balanced if

\( \forall (W, V) \in \mathcal{E}, \forall (\omega, \nu) \in [0,1]^2, T(\omega, \nu) = 1 \text{ iff } \exists \hat{\pi} \in [0,1]^{k-2}, \text{ an assignment to neighbors } \tilde{U} \text{ of } V \text{ other than } W, \)

5. 6.

1. \( T(\omega, U) = 1, \forall i = 1, \ldots, k-2 \)

2. \( V = \nu \text{ is BR}. \text{ So } W = \omega, \tilde{U} = \tilde{V}. \)

**Defn.** A joint mixed strategy \( \vec{\pi} \in [0,1]^m \) is consistent with balanced tables \( \{ T_{WV} \} \) if \( \forall (W,V) \in \mathcal{E}, T_{WV}(\vec{\pi}_W, \vec{\pi}_V) = 1. \)

**Observation:** The limit tables \( \{ T_{WV}^* \} \) are balanced.

* A characterization of Nash equilibria in a graphical game:

  A mixed strategy \( \vec{\pi} \) is a NE for the graphical game GG iff \( \vec{\pi} \) is consistent with the limit (balanced) tables for GG.
Approximate NE:

- As in ApproxTreeNash, discretize each player's mixed strategy space.

\[
\mathbb{M} = \{0, 1, 2, \ldots, \frac{1}{\varepsilon}\}
\]

- Griding size sufficiency results hold for arbitrary graph.
  So, set

\[
\frac{1}{\varepsilon} = \frac{\varepsilon}{2k}
\]

as before.

- Table size

\[
|\frac{1}{\varepsilon} T| = O\left(\left(\frac{2k}{\varepsilon}\right)^2\right)
\]

Poly in game representation size.

Running time (per round per player per neighbor):

\[
O\left(\left(2 \frac{1}{\varepsilon} T\right)^k\right) = O\left(\left(\frac{4k}{\varepsilon}\right)^k\right)
\]

[For \(k, s.t. k \log k = O(k \log n)\), poly in representation size]

\[
\uparrow \quad \text{max neighborhood size}
\]

\[
\uparrow \quad \text{# of players}
\]

- Convergence result \(\Rightarrow\) Table-passing phase converges in a finite # of rounds. So total running time

for this phase:

\[
O\left(\left|\mathcal{E}\right| \left(\frac{2k}{\varepsilon}\right)^2 \left(\frac{4k}{\varepsilon}\right)^k \left|\mathcal{E}\right|\right) = O\left(n^2 k^{k+1} 2^k \left(\frac{1}{\varepsilon}\right)^{k+2}\right)
\]

[For \(k, s.t. k \log(k)\), poly in model size]

- at each round, at least one table entry changes \([\text{from } 1 \text{ to } 0]\)

- We have \(|\mathcal{E}|\) tables, each of size \(O\left(\frac{1}{\varepsilon^2}\right)\).

[Recall, \(G\) undirected, so \((u, v) \in \mathcal{E} \iff (v, u) \in \mathcal{E}\]

[Easy to modify to directed case].
Observation: Lemma 6 of KLS. (union of rectangles representation) is general, so it applies here. Hence, we can do table-message passing exactly for 2-action games, but it might not converge in finite time!

- Table representation can grow exponentially with the # of rounds!

Assignment-passing phase

Basic Idea: Table-passing phase might have left us with a significantly smaller search space!

[See Accompanying PowerPoint presentation for an example of ideal behavior]

- A "generalization" of "upstream pass" in TreeNash

  BUT with a "wrinkle"

  *Defn: The projection set of player \( V \) is

  \[ \forall v \in [0,1], \ P_v(v) = 1 \text{ iff } \exists \ u \in [0,1]^{k-1}, \text{ s.t.} \]

  - \( T(v, u_i) = 1, \forall i = 1, \ldots, k-1 \)
  - \( V = v \) is BR to \( u = u \).

  [Alternative definitions exist]
Algorithm Sketch [for Assignment-passing phase]

Initialization:
- Pick arbitrary player $V$
- Select a mixed strategy $v \in P V(v) = 1$ [non-deterministically]

Iterate:
- At each round, in some sequential order over each player $V$
  - If $V$ has been set to $V$
    - If $V$ has unset neighbors,
      - Set unset neighbors s.t. resulting assignment to all neighbors is a witness
        [i.e., if $U$ is assignment to all neighbors,
        \[ T(v, u_i) = 1, \forall i = 1, \ldots, k - 1 \]
        \[ v = v \text{ is B.R. to } U = U \]
      - If not possible, backtrack!
    - Else, check for consistency
      - Backtrack if inconsistent
        - Neighborhood assignment

Remarks:
- For tree graphical games, no backtracking necessary!
- In general, need to backtrack to take care of inconsistencies
- Assignment-passing phase is worst-case exponential in # of players.

[See accompanying PowerPoint presentation for experimental results]