Complexity of Solving Strategic Games

The basic problem:

**NASH**: Given a finite 2-player strategic game $G$, find a mixed strategy profile $(\alpha, \beta)$ that is a NE of $G$ [if one exists, else return "no".]

In this form NASH looks similar to other search problems, e.g.:

**SAT**: Given a Boolean formula $\phi$ in CNF, find a truth assignment that makes $\phi$ true if one exists, else return "no".

Difference to SAT: existence of NE is guaranteed!

Search version of the usual decision problem
A search problem is given by a binary relation $R(x,y)$ over strings: Given $x$, find some $y$ such that $R(x,y)$ holds if such a $y$ exists, otherwise return "no".

Complexity classes for search problems:

$\overline{FP}$: class of search problems that can be solved by a deterministic Turing machine in polynomial time.

$\overline{FNP}$: ... (as above) ...

by a non-deterministic Turing machine ...

$\text{TFNP}$: class of search problems in $\overline{FNP}$ where the relation $R$ is known to be total, i.e. $\forall x \exists y R(x,y)$. 
PPAD: class of search problems that can be polynomially reduced to END-OF-LINE.

END-OF-LINE: Consider a directed graph with node set \( \{0, 1, \ldots, n\} \) such that each node has out- and in-degree at most 1. The graph is specified by two poly-time functions \( f \) and \( g \):

- \( f(v) \): successor candidate of \( v \) or empty
- \( g(v) \): predecessor candidate of \( v \) or empty

In the graph, there is an arc \( v \rightarrow v' \) if and only if \( f(v) = v' \) and \( g(v) = v' \).

Given a source node \( v \) in the graph, find some node \( v' \neq v \) such that \( v' \) has out-degree 0 or in-degree 0.
Notice:
* $FP \subseteq PPAD \subseteq TFNP \subseteq FNP$
* Lemke-Howson algorithm has exponential time complexity in the worst case.

**Theorem** (Daskalakis et al., 2006)

$NASH$ is $PPAD$-complete.

**2nd NASH:** Given a finite 2-player game $G$ and a NE of $G$, find a second NE of $G$ if one exists, else return "no."

**Theorem**

2nd NASH is $FNP$-complete.

**Problem:** Any $\subseteq$-relations proper?
Some further results: Given a finite 2-player game $G$, it is NP-hard to decide whether there exists a NE $(\alpha, \beta)$ in $G$ that has one of the following properties:

(a) player 1 (or 2) receives a payoff $\geq k$.

(b) $U_1(\alpha, \beta) + U_2(\alpha, \beta) \geq k$.

(c) $(\alpha, \beta)$ is Pareto-optimal, i.e., there is no strategy profile $(\alpha', \beta')$ such that

$U_i(\alpha', \beta') \geq U_i(\alpha, \beta)$ for both $i \in \{1, 2\}$, and

$U_i(\alpha', \beta') > U_i(\alpha, \beta)$ for at least one $i \in \{1, 2\}$.

(d) player 1 (or 2) plays some given action $a$ with probability $> 0$. 

\[ \rightarrow \text{Guaranteed payoff problem} \]

\[ \rightarrow \text{Guaranteed social welfare problem} \]
Extensive Games

So far: only simultaneous, one-shot games

Question: How to model the sequential structure of many games (e.g., chess...)?

Approach: Use extensive games (i.e., game trees)

Idea: Players have several choice points where they can decide how to play. Strategies, then, map choice points to applicable actions.
Definition: An extensive game with perfect information (EGWPI) is a tuple $\Gamma = (N, A, H, P, (u_i)_{i \in N})$ where:

- $N$ is a finite, nonempty set of players.
- $A$ is a nonempty set of actions.
- $H$ is a set of (finite or infinite) sequences over $A$ (called histories) such that:
  - the empty sequence $\langle \rangle \in H$;
  - if $\langle a_k \rangle_{k=1}^K \in H$ for some $K \in \mathbb{N} \cup \{\infty\}$ and $L < K$, then $\langle a_k \rangle_{k=L}^K \in H$;
  - if $\langle a_k \rangle_{k=1}^\infty$ is an action sequence such that $\langle a_k \rangle_{k=L}^\infty \in H$ for each $L \in \mathbb{N}$, then $\langle a_k \rangle_{k=1}^\infty \in H$.

Assumption:

All the ingredients of $\Gamma$ are common knowledge amongst the players at the same time.
A history is called **terminal** if it is infinite or if it is not the prefix of any longer history in \( \mathcal{H} \). The set of terminal histories is denoted by \( \mathcal{Z} \).

- \( \mathbf{P}: \mathcal{H} \times \mathcal{Z} \rightarrow \mathbb{N} \) is the \underline{player function} assigning to each non-terminal history \( h \in \mathcal{H} \setminus \mathcal{Z} \) a player \( \mathbf{P}(h) \) whose turn it is to "move" after \( h \).

- For each player \( i \in \mathbb{N} \), \( \mathbf{w}_i: \mathcal{Z} \rightarrow \mathbb{R} \) is player \( i \)'s \underline{utility function}.

Some terminology:
- \( \Gamma \) is \underline{finite} if \( \mathcal{H} \) is finite.
- \( \Gamma \) has \underline{finite horizon} if \( \mathcal{H} \) contains no infinite history.
Example (Shapley Game): Two players have to share two indistinguishable objects.

- Player 1 proposes an allocation.
- Player 2 accepts or declines the proposal.

Objects are allocated as proposed or no one gets anything.

Game tree:

Player 1's actions are proposed, and player 2's actions are accepted or declined. The payoffs are calculated as follows:

\[ u_1((2,0), y) = 2 \]
\[ u_2((1,1), u) = 0 \]
Formally, \( T = \langle N, A, H, P, (u_i)_{i \in N} \rangle \) where

- \( N = \{ 1, 2, 3 \} \)
- \( A = \{ (2, 0), (1, 1), (0, 2), y, a \} \)
- \( H = \{ \langle \rangle, \langle (2, 0) \rangle, \langle (1, 1) \rangle, \langle (0, 2) \rangle, \langle (2, 0), 2 \rangle, \langle (2, 0), m \rangle, \langle (1, 1), y \rangle \}, \ldots \} \)
- \( E = \{ h \in H : |h| = 2 \} \)
- \( \mathcal{P}(\langle \rangle) = 1, \mathcal{P}(h) = 2 \) for \( h \in H \setminus (E \cup \{ \langle \rangle \}) \)
- \( u_{a_n}(\langle (2, 0), y \rangle) = 2 \), etc.