Normal Form Games

- **Finite, \( n \)-person normal form game**: \((N, A, u)\):
  - \( N \) is a finite set of \( n \) players, indexed by \( i \)
  - \( A = A_1, \ldots, A_n \) is a set of actions for each player \( i \)
  - \( a \in A \) is an action profile
  - \( u = \{u_1, \ldots, u_n\} \), a utility function for each player, where \( u_i : A \rightarrow \mathbb{R} \)

Other types of games

- **Common payoff games**: a game where all action profiles \( a \in A_1 \times \ldots \times A_n \) for any pair of agents result in \( u_i(a) = u_j(a) \).
- **Constant sum games**: A game in which a constant \( c \) exists s.t. for each strategy profile \( a \in A_1 \times A_2 \) it is the case that \( u_1(a) + u_2(a) = c \).
Strategy

- **Pure strategy**: select an action and play it with positive probability.

- **Mixed strategy**: choose a set of random actions over a set of available actions based on some positive probability distribution.

Mixed Strategy

- Let \((N, A, u)\) be a normal form game, and for any set \(X\) let \(\Pi(X)\) be the set of all probability distributions over \(X\). Then the set of **mixed strategies** for player \(i\) is \(s_i = \Pi(A_i)\). The set of mixed strategy profiles is the Cartesian product of the individual mixed strategy sets, \(S_1 \times \ldots \times S_n\).
Mixed Strategy: Expected Utility

- Given a normal form game \((N, A, u)\) the expected utility \(u_i\) for player \(i\) of the mixed strategy profile \(s = (s_1, ... s_n)\) is defined as
  \[ u_i(s) = \sum_{a \in A} u_i(a) \prod_{j \neq i} s_j(a_j) \]

Best Response

- Player \(i\)’s best response to the strategy profile \(s_i\) is a mixed strategy \(s_i^* \in S_i\) s.t.
  \[ u_i(s_i^*, s_i) \geq u_i(s_i, s_i) \]
  for all strategies \(s_i \in S_i\)
  where
  \[ s_i = (s_1, ..., s_{i-1}, s_{i+1}, ..., s_n) \]
  \[ s = (s_i, s_i) \]
Nash Equilibrium

A strategy profile \( s = (s_1, \ldots, s_n) \) is a **Nash Equilibrium** if, for all agents \( i \), \( s_i \) is a best response to \( s_{-i} \).

- two strategies \( s_1 \) and \( s_2 \) are in Nash equilibrium if:
  1. under the assumption that agent \( i \) plays \( s_1 \), agent \( j \) can do no better than play \( s_2 \); and
  2. under the assumption that agent \( j \) plays \( s_2 \), agent \( i \) can do no better than play \( s_1 \).

- Every finite normal form game has a Nash Equilibrium.

Dominating strategies

Let \( s_i \) and \( s'_i \) be two strategies for agent \( i \), and \( S_{-i} \) the set of all strategy profiles of the remaining agents, then

- \( s_i \) **strictly dominates** \( s'_i \) if for all \( s_{-i} \in S_{-i} \) it is the case that \( u_i(s_i, s_{-i}) > u_i(s'_i, s_{-i}) \).

- \( s_i \) **weakly dominates** \( s'_i \) if for all \( s_{-i} \in S_{-i} \) it is the case that \( u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}) \) and for at least one \( s_{-i} \in S_{-i} \) it is the case that \( u_i(s_i, s_{-i}) > u_i(s'_i, s_{-i}) \).

- \( s_i \) **very weakly dominates** \( s'_i \) if for all \( s_{-i} \in S_{-i} \) it is the case that \( u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}) \).
Rationalizability

- A strategy is rationalizable if a perfectly rational player could justifiably playing it against one or more perfectly rational opponents.
  - Nash Equilibrium strategies are always rationalizable.

Pareto Optimality

- Sometimes one outcome $o$ is at least as good for every agent as another outcome $o'$, and there is some agent who strictly prefers $o$ to $o'$
  - we say that $o$ Pareto-dominates $o'$.
- An outcome $o$ is Pareto-optimal if there is no other outcome that Pareto-dominates it.
Pareto Optimality

- Given a normal-form game, a strategy profile $s$ is Pareto optimal (or efficient) if there does not exist another strategy profile $s'$ s.t. for all agents $i$ it is the case that $u_i(s_i) \leq u_i(s'_i)$, and for some agent $i$ it is the case that $u_i(s) < u_i(s')$.
- All strategy profiles in zero-sum games are Pareto efficient.
- Every Pareto outcome has the same payoff in a common-payoff game.

Correlated Equilibrium

- Given an $n$-agent game $G = (N, A, u)$ a correlated equilibrium is a tuple $(v, \pi, \sigma)$, where $v$ is a tuple of random variables $v = (v_1, ..., v_n)$ with respective domains $D = (D_1, ..., D_n)$, $\pi$ is a joint distribution over $v$, $\sigma = (\sigma_1, ..., \sigma_n)$ is a vector of mappings $\sigma_i: D_i \rightarrow A_i$ and for each agent $i$ and every mapping $\sigma'_i: D_i \rightarrow A_i$, it is the case that
  \[
  \sum_{d \in D} \pi(d)u_i(\sigma_1(d_1), ..., \sigma_i(d_i), ..., \sigma_n(d_n)) \geq \sum_{d \in D} \pi(d)u_i(\sigma'_1(d_1), ..., (\sigma'_i(d_i), ..., \sigma_n(d_n))
  \]
- For every NE an equivalent correlated equilibrium can be constructed.
\textbf{\textit{\text{ε-\text{-Nash Equilibrium}}}}

- A strategy profile \( s = (s_1, ..., s_n) \) is an \textit{ε-\text{-Nash equilibrium}} if, for all agents \( i \) and for all strategies \( s'_i \neq s_i \),
  \[ u_i(s_i, s_{-i}) \geq u_i(s'_i, s_{-i}) - \epsilon. \]
- \textit{ε-\text{-Nash equilibrium} always exist and are easier to compute than NE.}
- An \textit{ε-\text{-Nash equilibrium} is not necessarily close to a Nash equilibrium.}

\textbf{\textit{NE of two-player zero-sum games}}

\[
\begin{align*}
\text{minimize} & \quad U^*_1 \\
\text{subject to} & \quad \sum_{i \in i_1} u_i(a_i^1, a_{-i}^2) \cdot s_{-i}^2 + r_i^j = U^*_1 \quad \forall j \in A_1 \\
& \quad \sum_{i \in A_1} s_i = 1 \\
& \quad s_{-i}^2 \geq 0 \quad \forall k \in A_2 \\
& \quad r_i^j \geq 0 \quad \forall j \in A_1
\end{align*}
\]

- Introduce \textit{slack variables}, \( r_i^j \).
NE of two-player general-sum games

\[ \sum_{j \in A_1} u_1(a^j_1, a^k_2) \cdot s^k_2 + r^j_1 = U^*_1 \quad \forall j \in A_1 \]
\[ \sum_{k \in A_2} u_2(a^j_1, a^k_2) \cdot s^j_1 + r^k_2 = U^*_2 \quad \forall k \in A_2 \]
\[ \sum_{j \in A_1} s^j_1 = 1, \quad \sum_{k \in A_2} s^k_2 = 1 \]
\[ s^j_1 \geq 0, \quad s^k_2 \geq 0 \quad \forall j \in A_1, \quad \forall k \in A_2 \]
\[ r^j_1 \geq 0, \quad r^k_2 \geq 0 \quad \forall j \in A_1, \quad \forall k \in A_2 \]
\[ r^j_1 \cdot s^j_1 = 0, \quad r^k_2 \cdot s^k_2 = 0 \quad \forall j \in A_1, \quad \forall k \in A_2 \]

Complexity is PPAD.

NE of two-player general-sum games

- The **complementarity condition** requires that when an action is played by a particular player with positive probability, the corresponding slack variable must be zero.
  - Each slack variable represents the player’s incentive to deviate from the corresponding action.
  - In equilibrium, all strategies played with some probability must yield the same expected payoff.
NE of two-player general-sum games

- Lemke-Howson algorithm
  - Initialize:
    - \((s_1, s_2) \leftarrow (0, 0)\)
    - Find and \(s'_1 \in G_1\) st. \(s'_1\) is adjacent to 0
    - \(x \leftarrow 1\)
  - Repeat
    - \(s_x \leftarrow s'_x\)
    - let \(a_{ij}\) be the label that occurs in both \(s_1\) and \(s_2\)
    - Find and \(s'_x \in G_x\) st \(s'_x\) is adjacent to \(s_x\) and \(a_{ij} \notin L(s'_x)\)
    - \(x \leftarrow 3 - x\)
  - Until: \((s_1, s_2)\) is a completely labeled pair.

NE for \(n\)-player general-sum games

- Simplicial subdivision algorithms
  - Consider
    - The player’s best responses as a function from points on the simplex to other points on the simplex.
  - Scarf’s algorithm: locates the fixed points.
    - Add a variable that expresses the accuracy of the current iteration’s approximation.
    - Worst case complexity: exponential in the number of players and the number of digits of accuracy.
NE for $n$-player general-sum games

- Generalize Support Enumeration Method to $n$-player case
  - The feasibility program becomes non-linear.
  - Algorithm must accommodate multiple variables in the feasibility problem.
    - Use standard numerical techniques for non-linear optimization.
    - Reverse the lexicographic ordering between the size and balance of supports.

Dominant Strategies

- A strategy dominates another when the first strategy is always at least as good as the second, independent of the other players’ actions.
  - Iterative removal
    - Strictly dominant strategies: order does not matter.
    - Very weakly and weakly dominant strategies: removal order can have an affect.
      - Potentially remove some equilibria of the original game.
      - Potentially remove a larger set of strategies and result in a smaller game.
Domination by a Mixed Strategy

- Mixed strategies cannot be enumerated.
- Strict Domination
  - Requires a linear program.

\[
\begin{align*}
\text{Minimize} & \quad \sum_{j \in A} p_j \\
\text{Subject to} & \quad \sum_{j \in A} p_j u_i(a_j, a_{-i}) \geq u_i(s_i, a_{-i}) \quad \forall a_{-i} \in A_{-i} \\
& \quad p_j \geq 0 \quad \forall j \in A_i
\end{align*}
\]

Domination by a Mixed Strategy

- Very weakly domination uses a feasibility program.

\[
\begin{align*}
\sum_{j \in A} p_j u_i(a_j, a_{-i}) & \geq u_i(s_i, a_{-i}) \quad \forall a_{-i} \in A_{-i} \\
p_j & \geq 0 \quad \forall j \in A_i \\
\sum_{j \in A} p_j & = 1
\end{align*}
\]
Domination by a Mixed Strategy

- **Weak domination**

Maximize \( \sum_{a_i \in A_i} \left( \sum_{j \in A_j} a_j \cdot \left[ p_j \cdot u_i(a_j, a_{-i}) - u_i(s_i, a_{-i}) \right] \right) \)

Subject to \( \sum_{j \in A_j} p_j \cdot u_i(a_j, a_{-i}) \geq u_i(s_i, a_{-i}) \quad \forall a_{-i} \in A_{-i} \)

\( p_j \geq 0 \quad \forall j \in A_j \)

\( \sum_{j \in A_j} p_j = 1 \)

---

Correlated Equilibrium \( n \)-player
general-sum games

\( \sum_{a \in A} p(a) u_i(a) \geq \sum_{a \in A} p(a) u_i(a', a_{-i}) \quad \forall i \in N, \forall a', a_{-i} \in A_{-i} \)

\( p(a) \geq 0 \)

\( \sum_{a \in A} p(a) = 1 \)

Variables: \( p(a) \); constants: \( u_i(a) \)

Changing this program to find the NE requires the first constraint to be

\( \sum_{a \in A} u_i(a) \prod_{j \in A_j} p_j(a_j) \geq \sum_{a \in A} u_i(a', a_{-i}) \prod_{j \in A_j} p_j(a_j) \quad \forall i \in N, \forall a', a_{-i} \in A_{-i} \)
Extensive Form Games

- **Extensive form games** are a representation that explicitly specifies a sequence or temporal structure.
  - Two variants:
    - Perfect information extensive-form games
    - Imperfect-information extensive-form games

Perfect-Information Extensive Form Games

- A (finite) perfect-information game (in extensive form) is a tuple $G = (N, A, H, Z, \chi, \rho, \sigma, u)$, where
  - $N$ is a set of $n$ players
  - $A = (A_1, \ldots, A_n)$ is a set of actions for each player
  - $H$ is a set of non-terminal choice nodes
  - $Z$ is a set of terminal nodes, disjoint from $H$
  - $\chi : H \rightarrow 2^A$ is the action function
    - assigns to each choice node a set of possible actions
  - $\rho : H \rightarrow N$ is the player function
    - assigns to each non-terminal node a player $i \in N$ who chooses an action at that node
  - $\sigma : H \times A \rightarrow H \cup Z$ is the successor function
    - maps a choice node and an action to a new choice node or terminal node for all $h \in H$ and $a \in A$, if $\sigma(h, a_1) = \sigma(h, a_2)$ then $h_1 = h_2$ and $a_1 = a_2$
  - $u = (u_1, \ldots, u_n)$, where $u_i : Z \rightarrow \mathbb{R}$ is a utility function for player $i$ on the terminal nodes $Z$
Pure Strategy

- A pure strategy is a complete specification of which deterministic action to take at every node belonging to that player.

- Formally, let $G = (N, A, H, Z, \chi, \rho, \sigma, u)$, be a perfect-information extensive-form game. Then the pure strategies of player $i$ consist of the cross product

$$\prod_{h \in H, \rho(h) = h} \chi(h)$$

Nash Equilibrium

- Theorem: Every perfect information game in extensive form has a Pure Strategy Nash Equilibrium (PSNE).
Subgame Perfect Equilibrium

- Define subgame of $G$ rooted at $h$: the restriction of $G$ to the descendents of $H$.
- Define set of subgames of $G$: subgames of $G$ rooted at nodes in $G$.

- $s$ is a subgame perfect equilibrium of $G$ iff for any subgame $G'$ of $G$, the restriction of $s$ to $G'$ is a Nash equilibrium of $G'$.
- Since $G$ is its own subgame, every subgame perfect equilibrium (SPE) is a NE.
- This definition rules out “non-credible threats”.

Computing Equilibria

- General-sum n-player: backward induction
  
  **function** BackwardInduction (node $h$) **returns** $u(h)$
  
  if $h \in T$ then return $u(h)$ \{ $h$ is a terminal node \}
  
  best_util $\leftarrow -\infty$
  
  for all $a \in \chi(h)$ do \{ all actions available at node $h$ \}
  
  util_at_child $\leftarrow$ BackwardInduction($(h, a)$)
  
  if util_at_child($h$) $>$ best_util$_{p(h)}$ then
    best_util $\leftarrow$ util_at_child
  
  end if
  
  end for
  
  **return** best_util

  Worst case computation: Exponential
Computing Equilibria

- Zero-sum two-player games: minimax and alpha-beta
  - Recall that effectiveness of alpha-beta depends upon the order that nodes are considered.
  - Best case: $O(b^{m/2})$
  - Worst case (random ordering): $O(b^{3m/4})$

Imperfect-information extensive-form games

- Imperfect-information extensive form games include agents that need to act with partial or no knowledge of the actions taken by others, or even themselves.
- An imperfect-information games (in extensive form) is a tuple $(N, A, H, Z, \chi, \rho, \sigma, u, I)$, where
  - $(N, A, H, Z, \chi, \rho, \sigma, u)$ is a perfect-information extensive-form game, and
  - $I = (I_1, \ldots, I_n)$, where $I_i = (I_{i,1}, \ldots, I_{i,k_i})$ is an equivalence relation on $(h \in H: \rho(h) = i)$ with the property that $\chi(h) = \chi(h')$ and $\rho(h) = \rho(h')$ whenever there exists a $j$ for which $h \in I_{ij}$ and $h' \in I_{ij}$. 
Randomized Strategies

- There are two meaningfully different kinds of randomized strategies in imperfect information extensive form games.
  - **Mixed strategy**: randomize over pure strategies
  - **Behavioral strategy**: independent coin toss every time an information set is encountered

---

Perfect Recall

- Player $i$ has **perfect recall** in an imperfect-information game $G$ if for any two nodes $h$, $h'$ that are in the same information set for player $i$, for any path $h_0, a_0, h_1, a_1, \ldots, h_n, a_n, h$ from the root of the game to $h$ (where the $h_i$ are decision nodes and the $a_j$ are actions) and any path $h'_0, a'_0, h'_1, a'_1, \ldots, h'_m, a'_m, h'$ from the root to $h'$ it must be the case that:
  1. $n = m$
  2. For all $0 \leq j \leq n$, $h_j$ and $h'_j$ are in the same equivalence class for player $i$.
  3. For all $0 \leq j \leq n$, if $\rho(h_j) = i$ (that is, $h_j$ is a decision node of player $i$), then $a_j = a'_j$. 

---
Computing Equilibria: Sequence form

- Let $G$ be an imperfect-information game of perfect recall. The sequence-form representation of $G$ is a tuple $(N, \Sigma, g, C)$, where:
  - $N$ is a set of agents.
  - $\Sigma = (\Sigma_1, ..., \Sigma_n)$ where $\Sigma_i$ is the set of sequences available to agent $i$.
  - $g = (g_1, ..., g_n)$ where $g_i: \Sigma \rightarrow \mathbb{R}$ is the payoff function for agent $i$.
  - $C = (C_1, ..., C_n)$ where $C_i$ is a set of linear constraints on the realization probabilities of agent $i$.

Computing Equilibria: Sequence form – Imperfect Information Extensive Form

- A realization plan for player $i \in N$ is a function $r_i: \Sigma_i \rightarrow \mathbb{R}$ satisfying the following constraints.
  - $r_i(\emptyset) = 1$
  - $\sum_{r \in \Sigma_i} r_i(r) = r_i(seq_i(I)) \quad \forall I \in I_i$
  - $r_i(\sigma_i) \geq 0 \quad \forall \sigma_i \in \Sigma_i$
Best Responses: two-player - Imperfect Information Extensive Form

- Leverage sequence form representation and create a linear program.

Maximize \[ \sum_{\sigma_1 \in \Sigma_1} \sum_{\sigma_2 \in \Sigma_2} g_1(\sigma_1, \sigma_2) r_2(\sigma_2) r_1(\sigma_1) \]

Subject to \[ r_1(\emptyset) = 1 \]
\[ \sum_{\sigma_1 \in \Sigma_1} r_1(\sigma_1) = r_1(seq_1(I)) \quad \forall I \in I_1 \]
\[ r_1(\sigma_1) \geq 0 \quad \forall \sigma_1 \in \Sigma_1 \]

Dual Program

Maximize \[ v_0 \]

Subject to \[ v_{l_1(\sigma_1)} - \sum_{I \in I_1(\Sigma_1)} v_{I} \geq \sum_{\sigma_2 \in \Sigma_2} g_1(\sigma_1, \sigma_2) r_2(\sigma_2) \quad \forall \sigma_1 \in \Sigma_1 \]

Computing Equilibrium: two-player zero-sum - Imperfect Information Extensive Form

Maximize \[ v_0 \]

Subject to \[ v_{l_1(\sigma_1)} - \sum_{I \in I_1(\Sigma_1)} v_{I} \geq \sum_{\sigma_2 \in \Sigma_2} g_1(\sigma_1, \sigma_2) r_2(\sigma_2) \quad \forall \sigma_1 \in \Sigma_1 \]
\[ r_1(\emptyset) = 1 \]
\[ \sum_{\sigma_1 \in \Sigma_1} r_1(\sigma_1) = r_1(seq_1(I)) \quad \forall I \in I_2 \]
\[ r_1(\sigma_1) \geq 0 \quad \forall \sigma_1 \in \Sigma_1 \]
Computing Equilibrium: two-player general-sum - Imperfect Information Extensive Form

Formulate as a linear complementarity problem

\[ r_1(\emptyset) = 1 \quad r_2(\emptyset) = 1 \]
\[ \sum_{\sigma_1 \in \text{Ext}_1(I)} r_1(\sigma_1) = r_1(\text{seq}_1(I)) \quad \forall I \in I_1 \]
\[ \sum_{\sigma_2 \in \text{Ext}_2(I)} r_2(\sigma_2) = r_2(\text{seq}_2(I)) \quad \forall I \in I_2 \]
\[ r_i(\sigma_i) \geq 0 \quad \forall \sigma_i \in \Sigma_i \]
\[ r_2(\sigma_2) \geq 0 \quad \forall \sigma_2 \in \Sigma_2 \]

\[
\begin{align*}
    v_1^1(\sigma_1) - \sum_{I \in \text{Ext}_1(\sigma_1)} v_I^1 - \left( \sum_{\sigma_2 \in \Sigma_2} g_1(\sigma_1, \sigma_2) r_2(\sigma_2) \right) &\geq 0 \quad \forall \sigma_1 \in \Sigma_1 \\
    v_2^2(\sigma_2) - \sum_{I \in \text{Ext}_2(\sigma_2)} v_I^2 - \left( \sum_{\sigma_1 \in \Sigma_1} g_2(\sigma_1, \sigma_2) r_1(\sigma_1) \right) &\geq 0 \quad \forall \sigma_2 \in \Sigma_2 \\
    r_1(\sigma_1) \left[ v_1^1(\sigma_1) - \sum_{I \in \text{Ext}_1(\sigma_1)} v_I^1 - \left( \sum_{\sigma_2 \in \Sigma_2} g_1(\sigma_1, \sigma_2) r_2(\sigma_2) \right) \right] & = 0 \quad \forall \sigma_1 \in \Sigma_1 \\
    r_2(\sigma_2) \left[ v_2^2(\sigma_2) - \sum_{I \in \text{Ext}_2(\sigma_2)} v_I^2 - \left( \sum_{\sigma_1 \in \Sigma_1} g_2(\sigma_1, \sigma_2) r_1(\sigma_1) \right) \right] & = 0 \quad \forall \sigma_2 \in \Sigma_2
\end{align*}
\]
Infinitely Repeated Games

- Consider an infinitely repeated game translated into an extensive form game:
  - An infinite tree thus, payoffs cannot be attached to terminal nodes, nor can they be defined as the sum of the payoffs in the stage games.

- Given an infinite sequence of payoffs \( r_1, r_2, \ldots \) for player \( i \), the average reward of \( i \) is
  \[
  \lim_{k \to \infty} \frac{\sum_{j=1}^{k} r_j / k}{k}
  \]

- Given an infinite sequence of payoffs \( r_1, r_2, \ldots \) for player \( i \) and a discount factor \( \beta \) with \( 0 \leq \beta \leq 1 \), the future discounted rewards of \( i \) is
  \[
  \sum_{k=0}^{\infty} \beta^k r_j
  \]

Infinitely Repeated Games

- What is a pure-strategy in an infinitely-repeated game?
  - A choice of action at every decision point or an action at every stage game.
  - But results in an infinite number of actions!

- Some famous strategies:
  - **Tit-for-tat**: Start out cooperating. If the opponent defected, defect in the next round. Then go back to cooperation.
  - **Trigger**: Start out cooperating. If the opponent ever defects, defect forever.
Infinitely Repeated Games

- Consider any $n$-player game $G = (N, A_i, u_i)$ and any payoff vector $r = (r_1, r_2, \ldots, r_n)$.
  - Let $v_i = \min_{a \in A_i} \max_{s \in S_i} u_i(s,a)$.

- A payoff profile $r$ is **enforceable** if $r_i \geq v_i$.

- A payoff profile $r$ is **feasible** if there exist rational, non-negative values $\alpha_q$ such that for all $i$, we can express $r_i$ as
  \[ \sum_{a \in A_i} \alpha_q u_i(a) \]
  with $\sum_{a \in A} \alpha_a = 1$.

---

Infinitely Repeated Games

- **Folk Theorem:**
  - Consider any $n$-player game $G$ and any payoff vector $(r_1, r_2, \ldots, r_n)$.
    1. If $r$ is the payoff in any Nash equilibrium of the infinitely repeated $G$ with average rewards, then for each player $i$, $r_i$ is enforceable.
    2. If $r$ is both feasible and enforceable, then $r$ is the payoff in some Nash equilibrium of the infinitely repeated $G$ with average rewards.
Stochastic Games

- A stochastic game is a generalization of repeated games.
  - The game played at any iteration depends on the previous game played and on the actions taken by all agents in that game.

- A stochastic game is a generalized Markov decision process.
  - There are multiple players.
  - One reward function for each agent.
  - The state transition function and reward functions depend on the action choices of all players.

Stochastic Games

- A stochastic game is a tuple \((Q,N,A,P,R)\), where
  - \(Q\) is a finite set of states,
  - \(N\) is a finite set of \(n\) players,
  - \(A_i = A_1 \times \cdots \times A_n\), where \(A_i\) is a finite set of actions available to player \(i\).
  - \(P : Q \times A \times Q \to [0, 1]\) is the transition probability function; let \(P(q, a, \hat{q})\) be the probability of transitioning from state \(s\) to state \(\hat{q}\) after joint action \(a\),
  - \(R = r_1, \ldots, r_n\) where \(r_i : Q \times A \to R\) is a real-valued payoff function for player \(i\).
Stochastic Games: Strategy

- What is a pure strategy in a stochastic game?
  - Pick an action conditional on every possible history.
  - Mixtures over these deterministic, pure strategies are also possible.

- Some interesting restricted classes of strategies:
  - **Behavioral strategy**: \( s(h_t, a_{ij}) \) returns the probability of playing action \( a_{ij} \) for history \( h_t \).
    - The assumption is that mixing occurs at each history independently, not once at the beginning of the game.
  - **Markov strategy**: \( s_i \) is a behavioral strategy in which \( s_i(h_t, a_{ij}) = s_i(h'_t, a_{ij}) \) if \( q_t = q'_t \), where \( q_t \) and \( q'_t \) are the final states of \( h_t \) and \( h'_t \), respectively.
    - for a given time \( t \), the distribution over actions only depends on the current state.
  - **Stationary strategy**: \( s_i \) is a Markov strategy in which \( s_i(h_{1t}, a_{ij}) = s_i(h'_{2t}, a_{ij}) \) if \( q_{1t} = q'_{2t} \), where \( q_{1t} \) and \( q'_{2t} \) are the final states of \( h_{1t} \) and \( h'_{2t} \), respectively.
    - no dependence even on \( t \).

Stochastic Games: Equilibrium

- **Markov perfect equilibrium**: A strategy profile consisting of only Markov strategies and is a Nash equilibrium regardless of the starting state.
  - Analogous to subgame-perfect equilibrium.

- **Theorem**: Every \( n \)-player, general sum, discounted reward stochastic game has a Markov perfect equilibrium.
Stochastic Games: Equilibrium – Average Rewards

- **Irreducible stochastic game:**
  - Every strategy profile gives rise to an irreducible Markov chain over the set of games.
  - During the (infinite) execution of the stochastic game, each stage game is guaranteed to be played infinitely often—for any strategy profile.
  - Without this condition, the limit of the mean payoffs may not be defined.

- **Theorem:** Every 2-player, general sum, average reward, irreducible stochastic game has a Nash equilibrium.

---

Stochastic Games: Equilibrium

- **Theorem (Folk):** For every 2-player, general sum, irreducible stochastic game, and every feasible outcome with a payoff vector \( r \) that provides to each player at least his maxmin value, there exists a Nash equilibrium with a payoff vector \( r \). This is true for games with average rewards, as well as games with large enough discount factors (i.e. with players that are sufficiently patient).
Bayesian Games

- Generally speaking, a Bayesian game is a set of games that differ only in their payoffs, a common prior defined over them, and a partition structure over the games for each agent.
  - Can define using Information sets, chance moves, or epistemic types.

Bayesian Games: Epistemic Type

- Directly represent uncertainty over the utility function using the notion of epistemic type.

- A Bayesian game (epistemic types) is a tuple \((N, A, \Theta, p, u)\) where
  - \(N\) is a set of agents,
  - \(A = (A_1, \ldots, A_n)\), where \(A_i\) is the set of actions available to player \(i\),
  - \(\Theta = (1, \ldots, n)\), where \(i\) is the type space of player \(i\),
  - \(p : \Theta \to [0, 1]\) is the common prior over types,
  - \(u = (u_1, \ldots, u_n)\), where \(u_i : A \times \Theta \to \mathbb{R}\) is the utility function for player \(i\).
Bayesian Games: Strategies

- Assume epistemic type Bayesian games.

- **Pure strategy**: \( s_i : \Theta_i \rightarrow A_i \)
  - A mapping from every type agent \( i \) could have to the action he would play if agent \( i \) had that type.

- **Mixed strategy**: \( s_i : \Theta_i \rightarrow \Pi(A_i) \)
  - A mapping from \( i \)'s type to a probability distribution over agent \( i \)'s action choices.

- \( s_i(a_j|\theta_j) \)
  - The probability under mixed strategy \( s_i \) that agent \( j \) plays action \( a_j \) given that \( j \)'s type is \( \theta_j \).

Bayesian Games: Strategies

- Three meaningful notions of expected utility:
  - **ex-ante**
    - The agent knows nothing about anyone’s actual type.
  - **ex-interim**
    - An agent knows its own type, but not the types of the other agents.
  - **ex-post**
    - The agent knows all agents’ types.
Bayesian Games: Strategies

- The set of agent $i$’s best responses to mixed strategy profile $s_{-i}$ are given by

$$BR_i(s_{-i}) = \arg \max_{s_i} EU_i(s'_i, s_{-i})$$

- It may seem odd that $BR$ is calculated based on $i$’s ex-ante expected utility.
- However, write $EU_i(s)$ as $\sum_t p(t)EU_i(s|t)$ and observe that $EU_i(s'_i, s_{-i}| \theta_i)$ does not depend on strategies that $i$ would play if agent $i$’s type were not $\theta_i$.
- Thus, are in fact performing independent maximization of $i$’s ex-interim expected utility conditioned on each type that agent $i$ could have.

Bayesian Games: Nash Equilibrium

- A Bayes-Nash equilibrium is a mixed strategy profile $s$ that satisfies

$$\forall i \quad s_i \in BR_i(s_{-i})$$
Applications

- Games and game theory are underlying components of many aspects of market based multiple agent algorithms.
  - Bayesian games can be used to express agent preferences, determine utilities and payments.
  - Define negotiation protocols: auctions
  - Define problems: scheduling, coalitions
  - Represent domains