2. Representation of strategies in tree-form decision spaces

Link

---

0 - Introduction

- This lecture demonstrates how strategies are represented for the case of *tree-form* decision making.

- Two critical properties of a game representation:
  - The set of strategies is a convex and compact set.
  - Each player's utility function is multilinear.
  These two properties guarantee optimization methods can be used.

1 - Tree-Form Decision Making

- *Tree-form sequential decision process* (TFSDP) - Problem where the agent interacts with the environment in two ways:
  - *Decision points* - agent must select an action from a set of legal actions.
  - *Observation points* - agent observes a signal from the environment.

  Decision and observation points are structured sequentially into a tree.

  Note: It is assumed the agent isn't forgetful, so the tree never cycles back to a previously visited point.

- TFSDPs provide a general formalism for extensive-form games with perfect recall.
  (ex. poker, bridge, MDP)

- Summary of notation in TFSDPs:

  - - set of decision points.

  - - set of legal actions, at a decision point

  - - set of observation points.

  - - set of possible signals at observation point

  - - set of sequences defined as

  - - parent sequence of decision point , defined as the last sequence on the path from the root of the TFSDP to decision point .
  (null set if the agent doesn't act before)

- *Sequence* (or) - decision-action pair.
- *Parent sequence* () - the last decision point encountered before the decision point .

- Differences between TFSDPs and extensive-form games:
  - TFSDPs define the decision problem for a single agent.
  - Extensive-form games encode the dynamics of all agents involved.
  - EFG, decision nodes belong to players and observation nodes belong to a fictitious player, nature, who chooses a stochastic action.

- Similarities between TFSDPs and extensive-form games:
  - decision points = information sets
  - sequences = (information set, action) pairs.

2 - Strategies in Tree-Form Decision Making

- A strategy for an agent in a TFSDP specifies a distribution over the set of actions at each decision point .

2.1 - Behavioral Strategies

- *Behavioral strategy* - defines the probability for selecting an action given a decision point, for every sequence. Noted as a vector indexed over sequences,

  Major drawback of this representation: in order to determine the probability of reaching a given terminal state, we must compute the product of all actions on the path.

  This causes some expressions which depend on this probability to be non-convex.
  (ex. expected utility)

2.2 - Sequence-Form Strategies

- *Sequence-form representation* - defines the product of the probabilities of all actions at all decision points on the path from the root to the action at decision point .
  Noted as a vector over sequences,

Note: Is this the probability of taking a given action in a given state when at the root?

- Probability-mass-conservation constraints:
  - For (non-root decision points),
  - For (root decision points),

  Because of the probability-mass-conservation constraint, the set of valid sequence-form strategies (denoted) is convex.

2.3 - Deterministic and Randomized Sequence-Form Strategies

- *Deterministic strategies* - strategies that always select the same action at each decision point. Deterministic strategies are a subset of all possible strategies.

- In sequence-form representation, deterministic strategies are the subset of whose elements are all either or .

- *Reduced normal-form strategies* - the set of deterministic sequence-form strategies, in game theory.

- **Theorem:** the set of deterministic sequence-form strategies generates the polytope of sequence-form strategies, that is,

2.4 Bottom-up decomposition of the polytope of sequence-form strategies

- The sequence-form representation lends itself nicely to several common optimization procedures.

- One such procedure is *bottom-up* decomposition - work from the leaf nodes upward to the root, constructing the sequence-form representation at each node.

- This process is done using two operations which conserve convexity:

  - *Cartesian product* - the cartesian product of two sets yields a new set of all possible combinations of the input sets. .

  - *Convex hull* - the convex hull of a set yields the smallest set that contains all points of the input set. This is akin to taking a linear weighted average of the points in a set.

    Ex. . This is the set of points between and . The convex hull of a set with three points forms a triangle in 2D.

- Bottom-up construction rules:

  - *Leaf nodes* - where is a probability simplex and is the number of actions.

  - *Decision points* - convex hull of the child node strategies. The set of strategies rooted at a decision node is the linear combination of child strategies weighted by the probability of taking the actions that lead to each child.

– *Observation points* - cartesian product of the child node strategies. The set of strategies rooted at an observation node is the combination of the independent strategies at its child nodes.