CS 486 — Lecture 16: Intro to Decision Theory and Decision Networks

1 Decision Theory

- Combines probability theory and utility theory.
- How should an agent act in an uncertain world?
- Recall probability theory is what an agent should believe based on evidence, and utility theory is what the agent wants.
- In other words, an agent should choose the action that maximizes the expected utility.
- Unfortunately, it's easier said than done to get the *absolute best utility* — finding if the utility is the best is NP-hard most of the time, and requires checking into the future.
- Hence, we use expected utility.

2 Constructing a Decision Network

- Given the situation (robot puts on pads to lower damage, choose between a short or long route, short route may cause it to slip):
  - RV: A represents whether an accident occurs or not.
  - Decision variables/actions: S is whether it takes the short route, P is whether it puts on pads.
- We have 3 types of nodes:
  - Chance nodes — represent random variables (similar to a Bayesian network).
  - Decision nodes — represent actions (decision variables).
  - Utility nodes — represent the agent's utility function on states (the "happiness" in each state).

3 Evaluating a Decision Network

- How do we choose an action?
  1. Set evidence variables for the current state.
  2. For each possible value of a decision node:
     (a) Set decision node to value
     (b) Calculate posterior probability for parent nodes of the utility node
     (c) Calculate expected utility for the action
  3. Return the action with the highest expected utility.
- Often, we'll get expected utilities that are functions — we will define *policies* to select which choices to make based on the function's parameters (ie: if $q > 0.4$ then pick policy A, otherwise pick policy B...).

4 Variable Elimination

- In our original example, we can actually combine Short and Pads together.
- For a single-stage decision network:
  - We prune all the nodes that are not ancestors of the utility node.
  - Then, sum out all chance nodes.
  - For the single remaining factor, return the maximum value and the assignment that gives the maximum value.