Project #2
In project #1, most groups either
- Compared search strategies to find simple paths
- Created a two level system
- Subgoals
- Simple paths

For project #2:
- Same simulator
- Same format for specifying problems
- But / specify the problems
- You have two weeks to improve your code
- Anticipate hard problems
- Parse problems into subgoals
- Plan at level of subgoals
- Plan paths to satisfy each subgoal

Project #2 (cont)
Two weeks from today (April 10)
- Freeze your code
- I will release my state and goal files

You have one more week (until April 17) to:
- Test your system on my problems/goals
- Write a 4 page paper (max) describing the results
  - This is a challenge paper. No motivation or literature section is required.
  - However, it does need an introduction that summarizes what a reader should learn, a results section, an analysis of the results, and a conclusion.
- Prepare a 5 minute in-class presentation (April 17)
  - Every team member writes one separate column describing their contribution

A Medical Example
The “Asia” network:

Inference
We can calculate \( P(D=d) \)
We start with the joint distribution:

\[
P(v)P(s)P(t | v)P(l | s)P(b | s)P(a | t,l)P(x | a)P(d | a,b)
\]
Dealing with Evidence

We start by writing the factors:

\[ P(V)P(s)P(t | v)P(l | s)P(b | s)P(a | t, l)P(x | a)P(d | a, b) \]

Since we know that \( V = t \), we don’t need to eliminate \( V \)
Instead, we can replace the factors \( P(V) \) and \( P(T|V) \) with
\[ f_{p(V)} = P(V = t) \quad f_{p(T|V)}(T) = P(T | V = t) \]

These “select” the appropriate parts of the original factors given the evidence
Note that \( f_{p(V)} \) is a constant, and thus does not appear in elimination of other variables

Example (cont)

Need to eliminate: \( s, x, t, a, b \)

\[ P(V)P(s)P(t | v)P(l | s)P(b | s)P(a | t, l)P(x | a)P(d | a, b) \]

\[ f(t)P(s)P(l | s)P(b | s)P(a | t, l)P(x | a)P(d | a, b) \]

Summing on \( x \) results in a factor with two arguments \( f_s(b, l) \)
In general, result of elimination may be a function of several variables

Dealing with Evidence

How do we deal with evidence?

Suppose get evidence \( P(L) \)
And want to compute \( P(L | V = t, S = f, D = t) \)

\[ \frac{P(L, V = t, S = f, D = t)}{P(V = t, S = f, D = t)} \]
Variable Elimination

Compute the probability of \( X_k \) given values to evidence variables \( E \)

\[
P(X_k, E) = \sum_{\text{all non-query, non-evidence variables}} P(X_k|P(X_i)) \prod P(X_i|P(X_i))
\]

Algorithm is same as before, with no need to perform summation with respect to evidence variables.

Complexity of inference

Theorem:
Computing \( P(X=x) \) in a Bayesian network is NP-hard

Approaches to inference

- **Exact inference**
  - Variable elimination
  - Join tree algorithm (not covered)

- **Approximate inference**
  - Simplify the structure of the network to make exact inference efficient (variational methods, loopy belief propagation) (not covered)

- **Probabilistic methods**
  - Stochastic simulation / sampling methods
  - Markov chain Monte Carlo methods

Inference by sampling

Suppose we can sample instances \( \langle X_1, \ldots, X_n \rangle \) according to \( P(X_1, \ldots, X_n) \)

Want to compute \( P(e) \)

The probability that a random sample \( \langle X_1, \ldots, X_n \rangle \) satisfies \( e \) is approximately \( P(e) \)

We can view each sample as tossing a biased coin with probability \( P(e) \) of “Heads”

Sampling a Bayesian Network

If \( P(X_1, \ldots, X_n) \) is represented by a Bayesian network, can we efficiently sample from it?

Idea: sample according to structure of the network
- Write distribution using the chain rule, and then sample each variable given its parents

BN sampling

Samples:
Let $X_1, \ldots, X_n$ be order of variables consistent with arc direction.

for $i = 1, \ldots, n$ do
    sample $x_i$ from $P(X_i \mid Pa(X_i))$
    (Note: since $Pa(X_i) \subseteq \{X_1, \ldots, X_{i-1}\}$, we already assigned values to them)
return $x_1, \ldots, x_n$

Sampling a complete instance is linear in number of variables
- Regardless of structure of the network

However, if $P(e)$ is small, we need many samples to get a decent estimate.
Can we sample from $P(X_1, ..., X_n | e)$?

If evidence is in roots of network, easily

If evidence is in leaves of network, we have a problem
• Our sampling method proceeds according to order of nodes in graph

Rejection sampling: keep those instantiations that are consistent with the values of the evidence variables

Estimate $P(X | e)$ by $N(X, e) / N(e)$ where $N(.)$ counts the number of times an event was sampled.