

Reasoning Over Time

[RN2] Sec 15.1-15.3, 15.5
[RN3] Sec 15.1-15.3, 15.5

CS 486/686
University of Waterloo
Lecture 12: Oct 18, 2012

Outline

- Reasoning under uncertainty **over time**
- Hidden Markov Models
- Dynamic Bayesian Networks

Static Inference

- So far...
 - Assume the world doesn't change
 - **Static probability distribution**
 - Ex: when repairing a car, whatever is broken remains broken during the diagnosis
- But the world evolves over time...
 - How can we use probabilistic inference for weather predictions, stock market predictions, patient monitoring, etc?

3

CS486/686 Lecture Slides (c) 2012 P. Poupart

Dynamic Inference

- Need to reason **over time**
 - Allow the world to evolve
 - Set of states (encoding all possible worlds)
 - Set of time-slices (snapshots of the world)
 - Different probability distribution over states at each time slice
 - Dynamics encoding how distributions change over time

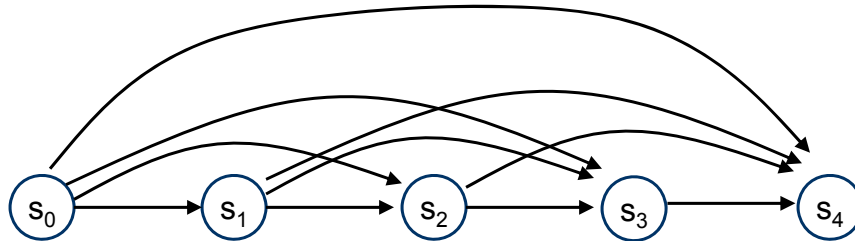
4

CS486/686 Lecture Slides (c) 2012 P. Poupart

Stochastic Process

- Definition

- Set of States: S
- Stochastic dynamics: $\Pr(s_t | s_{t-1}, \dots, s_0)$



- Can be viewed as a Bayes net with one random variable per time slice

5

CS486/686 Lecture Slides (c) 2012 P. Poupart

Stochastic Process

- Problems:

- Infinitely many variables
- Infinitely large conditional probability tables

- Solutions:

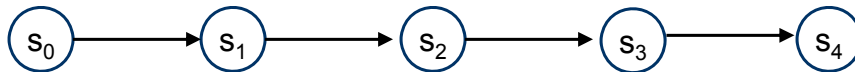
- **Stationary process**: dynamics do not change over time
- **Markov assumption**: current state depends only on a finite history of past states

6

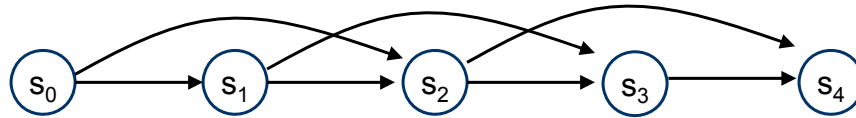
CS486/686 Lecture Slides (c) 2012 P. Poupart

K-order Markov Process

- Assumption: last k states sufficient
- First-order Markov Process
 - $\Pr(s_t | s_{t-1}, \dots, s_0) = \Pr(s_t | s_{t-1})$



- Second-order Markov Process
 - $\Pr(s_t | s_{t-1}, \dots, s_0) = \Pr(s_t | s_{t-1}, s_{t-2})$

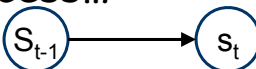


CS486/686 Lecture Slides (c) 2012 P. Poupart

7

K-order Markov Process

- Advantage:
 - Can specify entire process with **finitely many time slices**
- Two slices sufficient for a first-order Markov process...

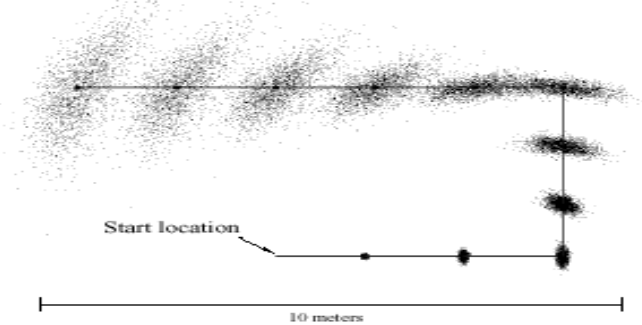
- Graph: 
- Dynamics: $\Pr(s_t | s_{t-1})$
- Prior: $\Pr(s_0)$

CS486/686 Lecture Slides (c) 2012 P. Poupart

8

Mobile Robot Localisation

- Example of a first-order Markov process



- Problem: uncertainty grows over time...

9

CS486/686 Lecture Slides (c) 2012 P. Poupart

Hidden Markov Models

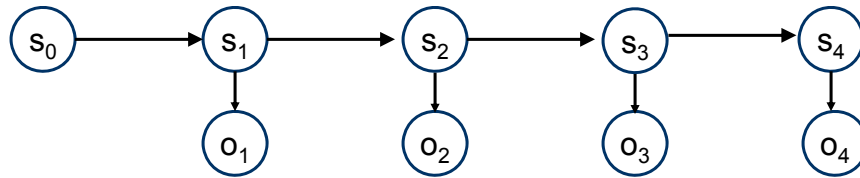
- Robot could use sensors to reduce location uncertainty...
- In general:
 - **States** not directly observable, hence uncertainty captured by a distribution
 - **Uncertain dynamics** increase state uncertainty
 - **Observations** made via sensors reduce state uncertainty
- Solution: **Hidden Markov Model**

10

CS486/686 Lecture Slides (c) 2012 P. Poupart

First-order Hidden Markov Model

- Definition:
 - Set of states: S
 - Set of observations: O
 - Transition model: $\Pr(s_t | s_{t-1})$
 - Observation model: $\Pr(o_t | s_t)$
 - Prior: $\Pr(s_0)$



CS486/686 Lecture Slides (c) 2012 P. Poupart

11

Mobile Robot Localisation

- (First-order) Hidden Markov Model:
 - S : (x,y) coordinates of the robot on a map
 - O : distances to surrounding obstacles (measured by laser range finders or sonars)
 - $\Pr(s_t | s_{t-1})$: movement of the robot with uncertainty
 - $\Pr(o_t | s_t)$: uncertainty in the measurements provided by laser range finders and sonars
- **Localisation** corresponds to the query:
 $\Pr(s_t | o_t, \dots, o_1)$?

CS486/686 Lecture Slides (c) 2012 P. Poupart

12

Inference in temporal models

- Four common tasks:
 - **Monitoring:** $\Pr(s_t | o_t, \dots, o_1)$
 - **Prediction:** $\Pr(s_{t+k} | o_t, \dots, o_1)$
 - **Hindsight:** $\Pr(s_k | o_t, \dots, o_1)$ where $k < t$
 - **Most likely explanation:**
 $\operatorname{argmax}_{s_t, \dots, s_1} \Pr(s_t, \dots, s_1 | o_t, \dots, o_1)$
- What algorithms should we use?
 - First 3 tasks can be done with variable elimination and 4th task with a variant of variable elimination

13

CS486/686 Lecture Slides (c) 2012 P. Poupart

Monitoring

- $\Pr(s_t | o_t, \dots, o_1)$: distribution over current state given observations
- Examples: robot localisation, patient monitoring
- **Forward algorithm:** corresponds to variable elimination
 - Factors: $\Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t$
 - Restrict o_1, \dots, o_t to the observations made
 - Summout s_0, \dots, s_{t-1}
 - $\sum_{s_0 \dots s_{t-1}} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i)$

14

CS486/686 Lecture Slides (c) 2012 P. Poupart

Prediction

- $\Pr(s_{t+k} | o_t, \dots, o_1)$: distribution over future state given observations
- Examples: weather prediction, stock market prediction
- **Forward algorithm**: corresponds to variable elimination
 - Factors: $\Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t+k$
 - Restrict o_1, \dots, o_t to the observations made
 - Summout $s_0, \dots, s_{t+k-1}, o_{t+1}, \dots, o_{t+k}$
 - $\sum_{s_0 \dots s_{t+k-1}, o_{t+1} \dots o_{t+k}} \Pr(s_0) \prod_{1 \leq i \leq t+k} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i)$

CS486/686 Lecture Slides (c) 2012 P. Poupart

Hindsight

- $\Pr(s_k | o_t, \dots, o_1)$ for $k < t$: distribution over a past state given observations
- Example: crime scene investigation
- **Forward-backward algorithm**: corresponds to variable elimination
 - Factors: $\Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t$
 - Restrict o_1, \dots, o_t to the observations made
 - Summout $s_0, \dots, s_{k-1}, s_{k+1}, \dots, s_t$
 - $\sum_{s_0 \dots s_{k-1}, s_{k+1}, \dots, s_t} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i)$

16

CS486/686 Lecture Slides (c) 2012 P. Poupart

Most likely explanation

- $\text{Argmax}_{s_0 \dots s_t} \Pr(s_0, \dots, s_t | o_t, \dots, o_1)$: most likely state sequence given observations
- Example: speech recognition
- **Viterbi algorithm**: corresponds to a variant of variable elimination
 - Factors: $\Pr(s_0), \Pr(s_i | s_{i-1}), \Pr(o_i | s_i), 1 \leq i \leq t$
 - Restrict o_1, \dots, o_t to the observations made
 - Maxout s_0, \dots, s_t
 - $\max_{s_0 \dots s_t} \Pr(s_0) \prod_{1 \leq i \leq t} \Pr(s_i | s_{i-1}) \Pr(o_i | s_i)$

17

CS486/686 Lecture Slides (c) 2012 P. Poupart

Complexity of temporal inference

- Hidden Markov Models are Bayes nets with a polytree structure
- Hence, variable elimination is
 - Linear w.r.t. to # of time slices
 - Linear w.r.t. to largest conditional probability table ($\Pr(s_t | s_{t-1})$ or $\Pr(o_t | s_t)$)
- **What if # of states or observations are exponential?**

18

CS486/686 Lecture Slides (c) 2012 P. Poupart

Dynamic Bayesian Networks

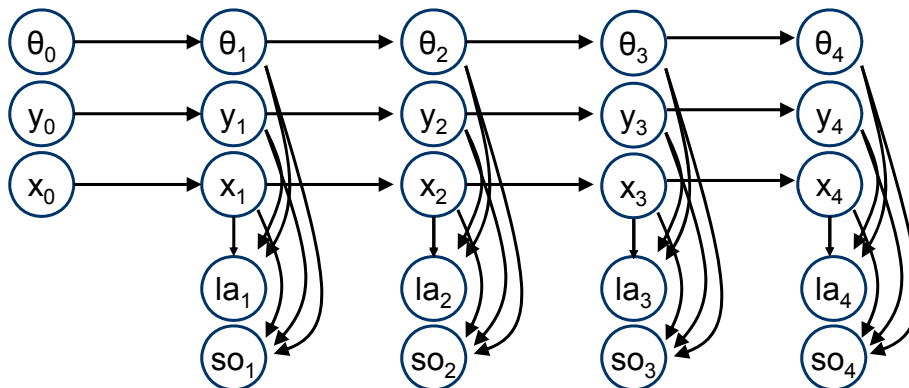
- Idea: **encode states and observations with several random variables**
- Advantage: exploit conditional independence to save time and space
- HMMs are just DBNs with one state variable and one observation variable

19

CS486/686 Lecture Slides (c) 2012 P. Poupart

Mobile Robot Localisation

- States: (x,y) coordinates and heading θ
- Observations: laser and sonar



20

CS486/686 Lecture Slides (c) 2012 P. Poupart

DBN complexity

- Conditional independence allows us to write transition and observation models **very compactly!**
- Time and space of inference: conditional independence rarely helps...
 - inference tends to be exponential in the number of state variables
 - Intuition: all state variables eventually get correlated
 - **No better than with HMMs** ☹️

21

CS486/686 Lecture Slides (c) 2012 P. Poupart

Non-Stationary Process

- What if the process is not stationary?
- Solution: add new state components until dynamics are stationary
- Example:
 - Robot navigation based on (x, y, θ) is non-stationary when velocity varies...
 - Solution: add velocity to state description e.g. (x, y, v, θ)
 - If velocity varies... then add acceleration
 - Where do we stop?

22

CS486/686 Lecture Slides (c) 2012 P. Poupart

Non-Markovian Process

- What if the process is not Markovian?
- Solution: add new state components until dynamics are Markovian
- Example:
 - Robot navigation based on (x,y,θ) is non-Markovian when influenced by battery level...
 - Solution: add battery level to state description e.g. (x,y,θ,b)

23

CS486/686 Lecture Slides (c) 2012 P. Poupart

Markovian Stationary Process

- Problem: adding components to the state description to force a process to be Markovian and stationary may significantly **increase computational complexity**
- Solution: try to find the smallest state description that is self-sufficient (i.e., Markovian and stationary)

24

CS486/686 Lecture Slides (c) 2012 P. Poupart

Probabilistic Inference

- Applications of static and temporal inference are virtually limitless
- Some examples:
 - mobile robot navigation
 - speech recognition
 - patient monitoring
 - help system under Windows
 - fault diagnosis in Mars rovers
 - etc.

25

CS486/686 Lecture Slides (c) 2012 P. Poupart

Next Class

- Markov Decision Processes

26

CS486/686 Lecture Slides (c) 2012 P. Poupart