CS 3630!

Lecture 8:
A Vacuum Cleaning Robot: Sensing, and Perception

Sensing

- For the trash sorting robot, we had multiple sensors, and their measurements were conditionally independent (given state).
- We could combine those measurements using Bayes to formulate state estimates.
- For the vacuum cleaning robot, we'll use a single sensor that has only three possible outputs: ***not very powerful***.
- We'll take measurements at each time step, and combine these with the robot's knowledge about its actions and the environment to make inferences about state.
- Bayes networks – and various special cases of Bayes nets – will be the key inference tool.

Trash Sorting Sensors

- Three sensors (weight, conductivity, vision-classifier).
- At any time t, collect measurements from the three sensors: z_t^w, z_t^c, z_t^v and use Bayes to compute $P(X_t = x \mid z_t^w, z_t^c, z_t^v)$.
- Measurements are conditionally independent given state, which gives a nice computational simplification after applying Bayes.

➢ The passing of time was irrelevant – each new sensor measurement was for a new piece of trash:

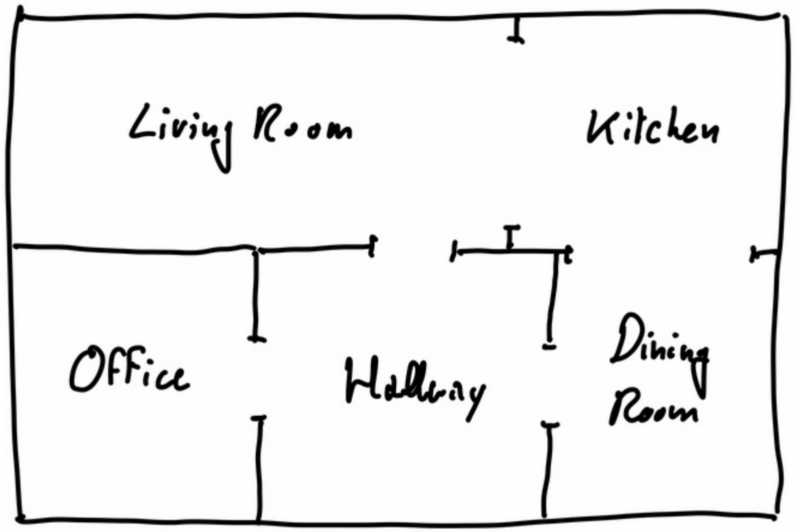
  - Completely independent of previous measurements
  - Completely independent of previous actions
  - Completely independent of previous states

This is not the case for our vacuuming robot!

Vacuuming robot sensor

- A single sensor that detects light levels, and returns a measurement z:
  - Bright, $z = 2$
  - Medium, $z = 1$
  - Dark, $z = 0$

X1	dark	medium	light
Living Room	0.1	0.1	0.8
Kitchen	0.1	0.1	0.8
Office	0.2	0.7	0.1
Hallway	0.8	0.1	0.1
Dining Room	0.1	0.8	0.1



- Sun is to the south, so plenty of light for living room and kitchen.
- Office and Dining room are poorly lit via windows.
- Hallway has no windows and is always dark.

- ***For Hallway, $(z = 0\,|H) = 0.8$, MLE will do the job!***
- ***For $z = 1, z = 2$, there's really no way to uniquely identify state from one measurement.***

Exploiting History

- Suppose we observe a sequence of measurements and actions:

$$z_1 = 0, a_1 = up, z_2 = 2$$

➢ It seems likely that $x_1 = H, x_2 = L$

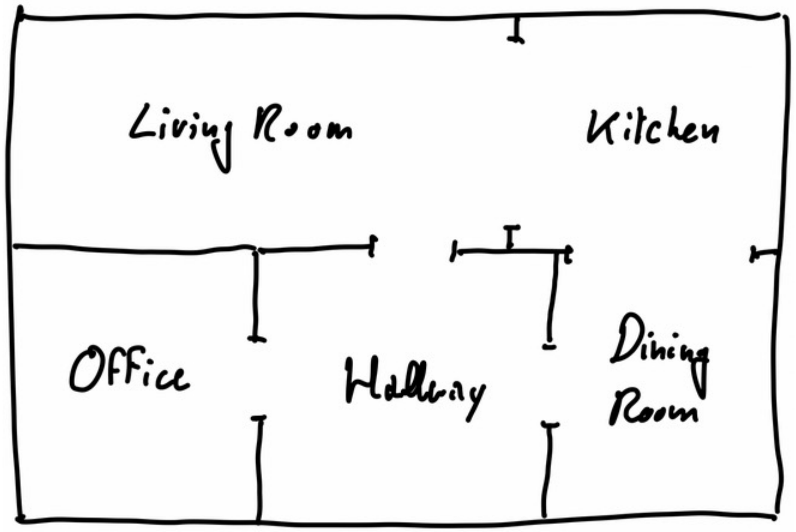
- Suppose we observe a sequence of measurements and actions:

$$z_1 = 1, a_1 = right$$
$$z_2 = 0, a_2 = right$$
$$z_3 = 1$$

➢ It seems likely that $x_1 = O, x_2 = H, x_3 = D$

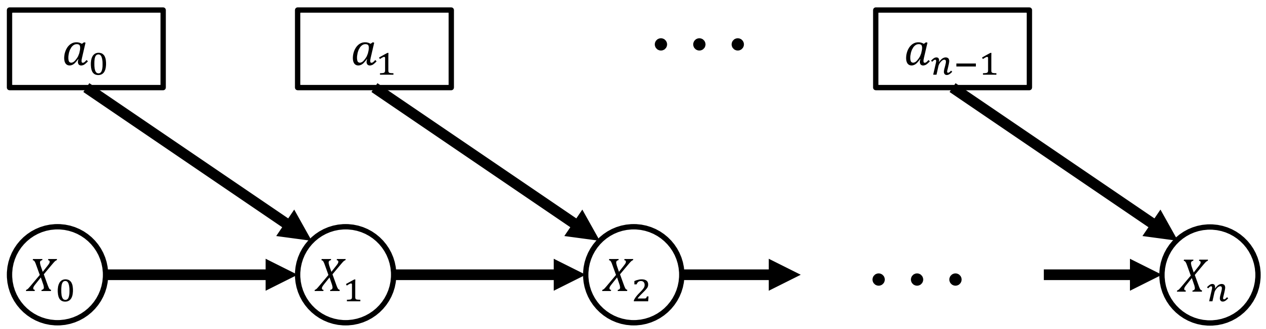
These examples illustrate the basic idea, but these examples are really simple.

How do we formalize/generalize this into a sensor model that accounts for actions and measurements as time sequences?



Bayes Networks

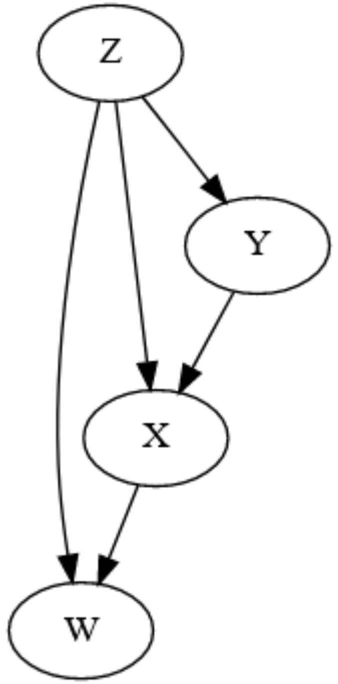
In the past, we have seen graphical models for various sorts of Markov chains:



These models are special cases of the more general Bayesian Networks (Bayes nets):

- Directed Acyclic Graph (DAG)
- For conditional probability $P(X|Y_1, \dots, Y_m)$ there are directed edges from each of Y_i to X.
- There are no other edges in the graph.

Bayes Nets



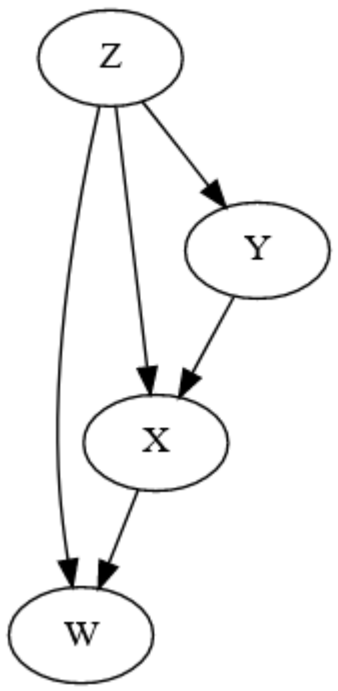
This network represents several conditional probability relationships:

- $P(W|X,Z)$
- $P(X|Y,Z)$
- $P(Y|Z)$
- $P(Z)$

Perhaps more importantly, Bayes nets explicitly encode ***conditional independence*** relationships:

- W is conditionally independent of Y given X

The (first) Magic of Bayes Nets



For a Bayes net with variables $X_1 \dots X_n$, the joint distribution is given by:

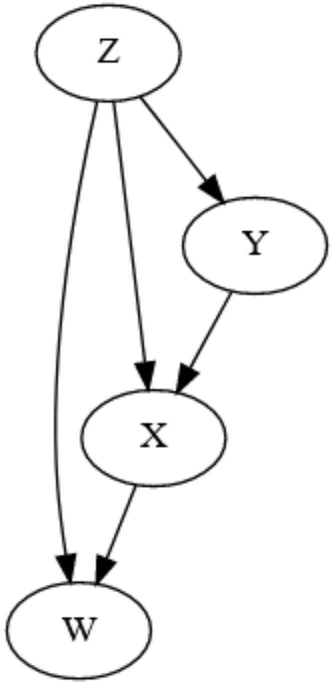
$$\boldsymbol{P(X_1 \dots X_n) = \prod_i P(X_i | \mathcal{P}(X_i))}$$

where $\boldsymbol{\mathcal{P}(X_i)}$ denotes the set of parents of node $\boldsymbol{X_i}$

For this specific network, the joint distribution is given by

$$\boldsymbol{P(W, X, Y, Z) = P(W|X, Z) P(X|Y, Z) P(Y|Z) P(Z)}$$

The (first) Magic of Bayes Nets



We can see why this works (for this example) by expanding the chain rule for joint probability distributions:

$$\boldsymbol{P(W,X,Y,Z) = P(W|X,Y,Z)P(X|Y,Z)P(Y|Z)P(Z)}$$

But from the topology of the Bayes net, we know

$$\boldsymbol{P(W|X,Y,Z) = P(W|X,Z)}$$

Making this substitution, we arrive to the desired result:

$$\boldsymbol{P(W,X,Y,Z) = P(W|X,Z)P(X|Y,Z)P(Y|Z)P(Z)}$$

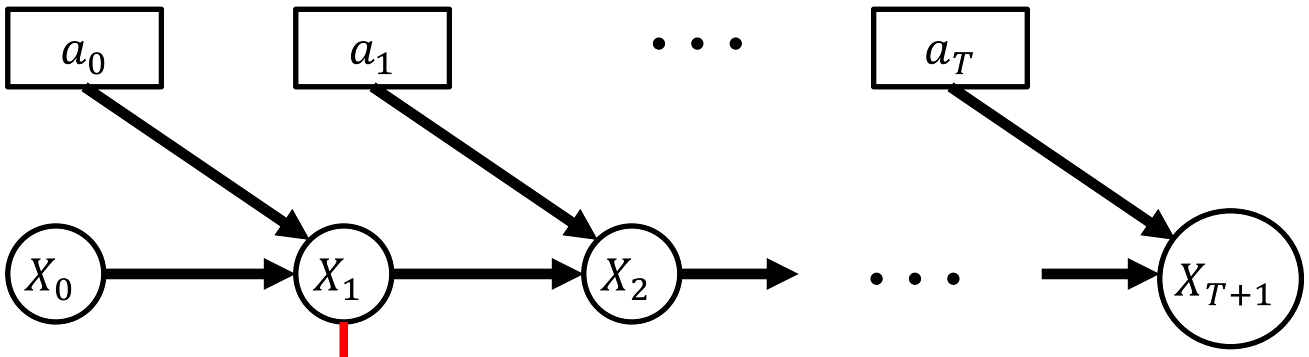
More Magic of Bayes Nets

How difficult would it be to explicitly encode the joint distributions for our vacuum cleaning robot?

Suppose we consider $X_1, \dots X_{T+1}$, and we want to encode $P(X_1, \dots X_{T+1})$

X_1	X_2	$\cdots$	X_T	X_{T+1}	$P(X_1, \dots X_{T+1})$
L	L		L	L	
L	L		L	K	
			L	O	
			L	H	
			L	D	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Number of rows = $|\mathcal{X}|^{T+1}$

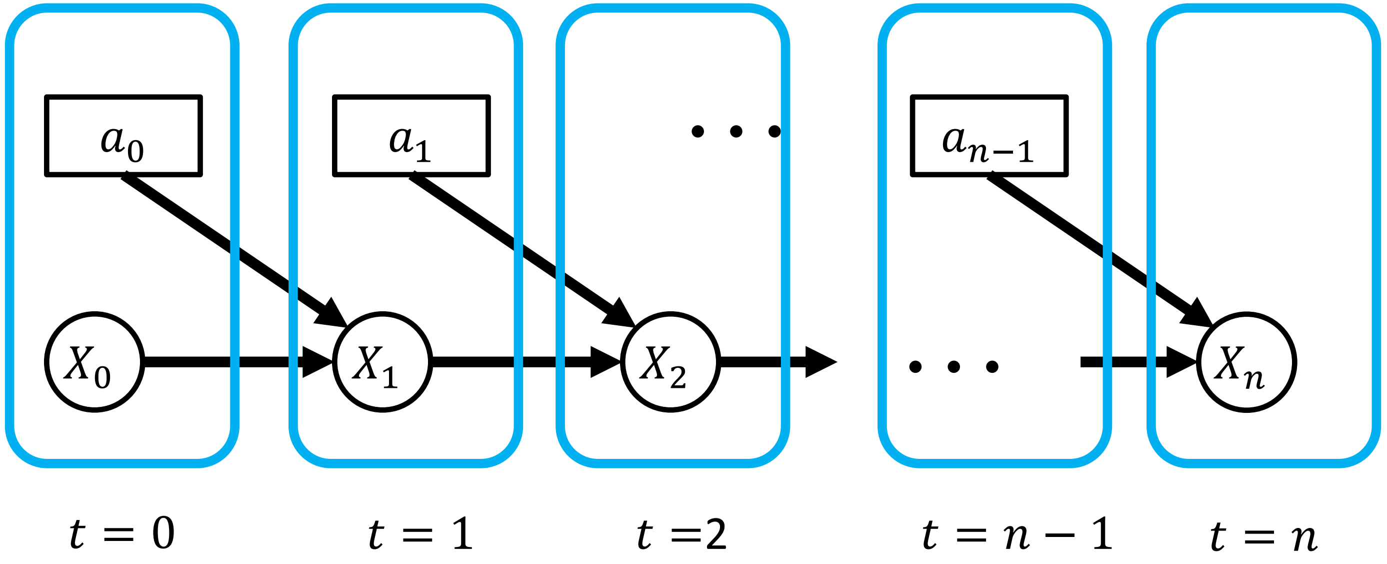


rows in CPT = $|\mathcal{X}|^2 \times |\mathcal{A}|$

Total storage $\approx (\mathrm{T}+1)(|\mathcal{X}|^2 \times |\mathcal{A}|)$

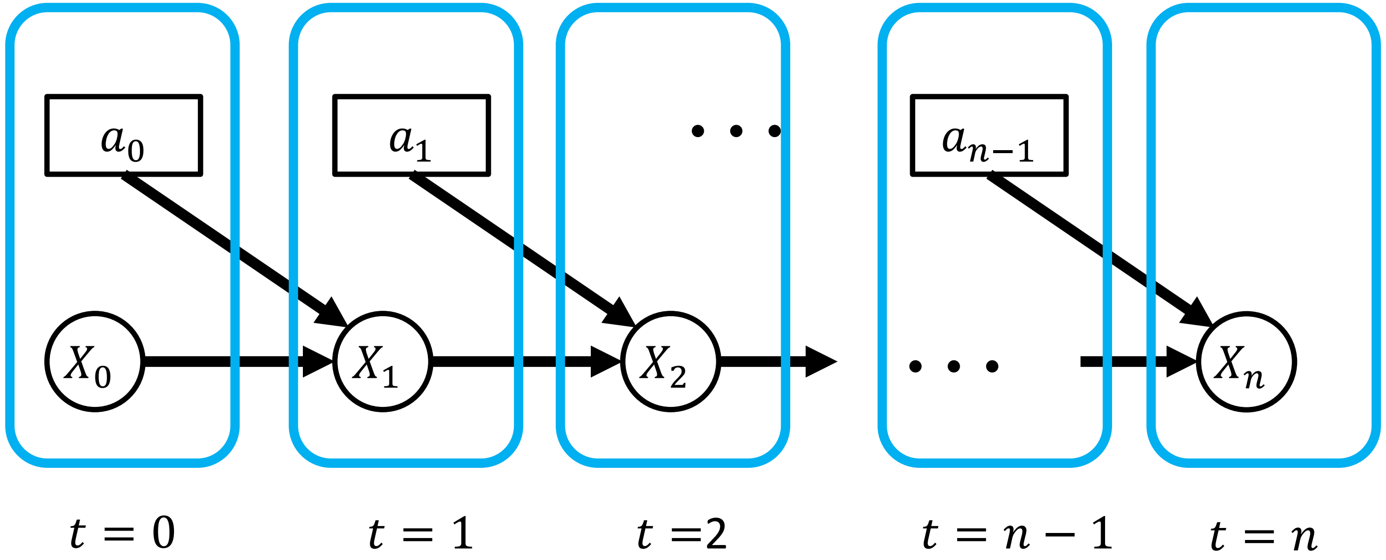
Dynamic Bayes Nets

- Bayes nets can be used to represent systems that evolve over time.
- Our vacuum cleaning robot is an example of such a system, at any time t, we have x_t and a_t and together, these determine (probabilistically) what happens for x_{t+1}.
- A dynamic Bayes net has a simple structure that repeats at each time step:



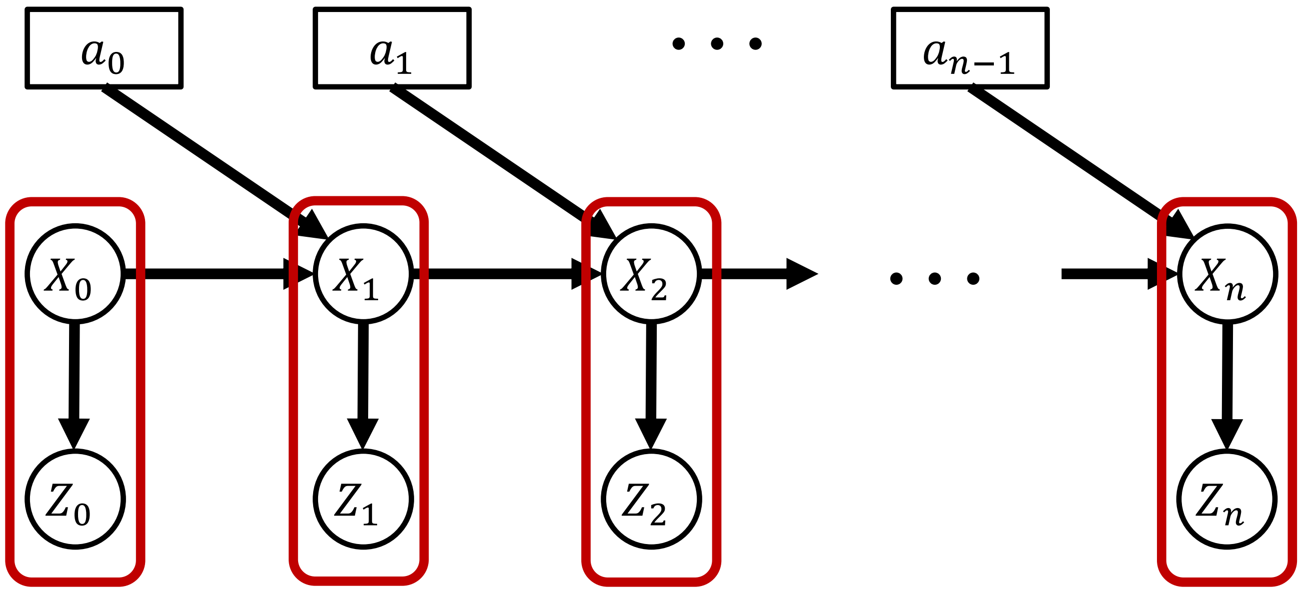
Simulation

- Forward simulation is easy for Dynamic Bayes Nets (DBNs).
- Sample initial state x_0 from the prior $P(X_0)$
- For each k generate a sample x_{k+1} from the distribution $P(X_{k+1}|X_k = x_k, a_k)$
- This is sometimes called ancestral sampling: to generate a sample for some node, look at its immediate ancestors.



Observations

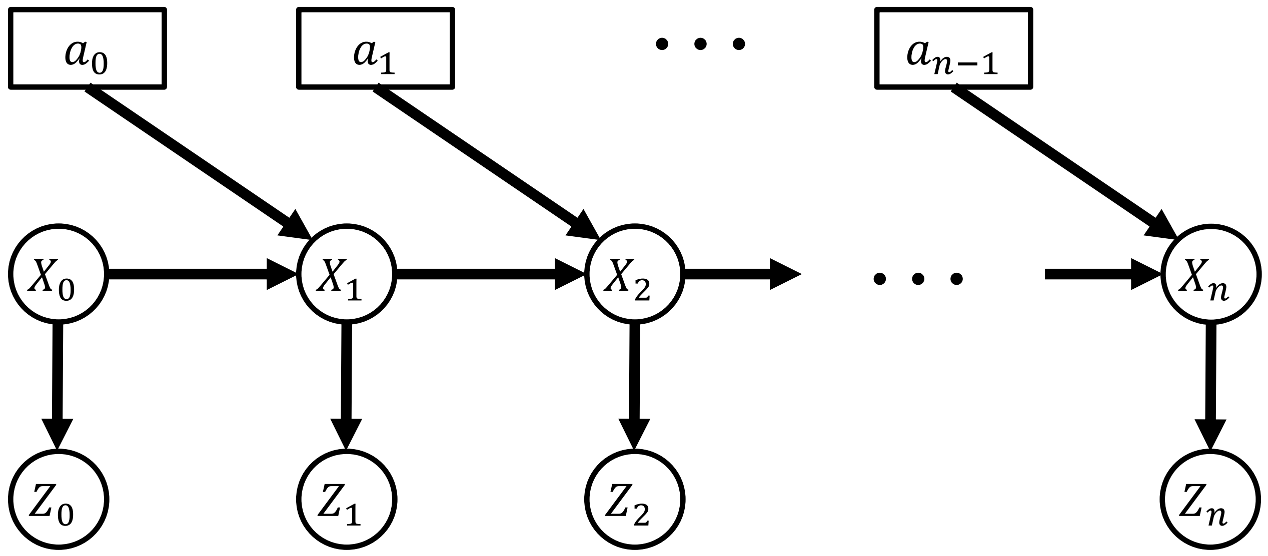
- The motivation for all of this Bayes net machinery was the idea that the history of sensor measurements was interesting. How do we encode this in a Bayes net?
- Recall our sensor model: $\boldsymbol{P(Z_t | X_t)}$.
- **This is easy to encode in a Bayes net!**



Still More Magic of Bayes Nets

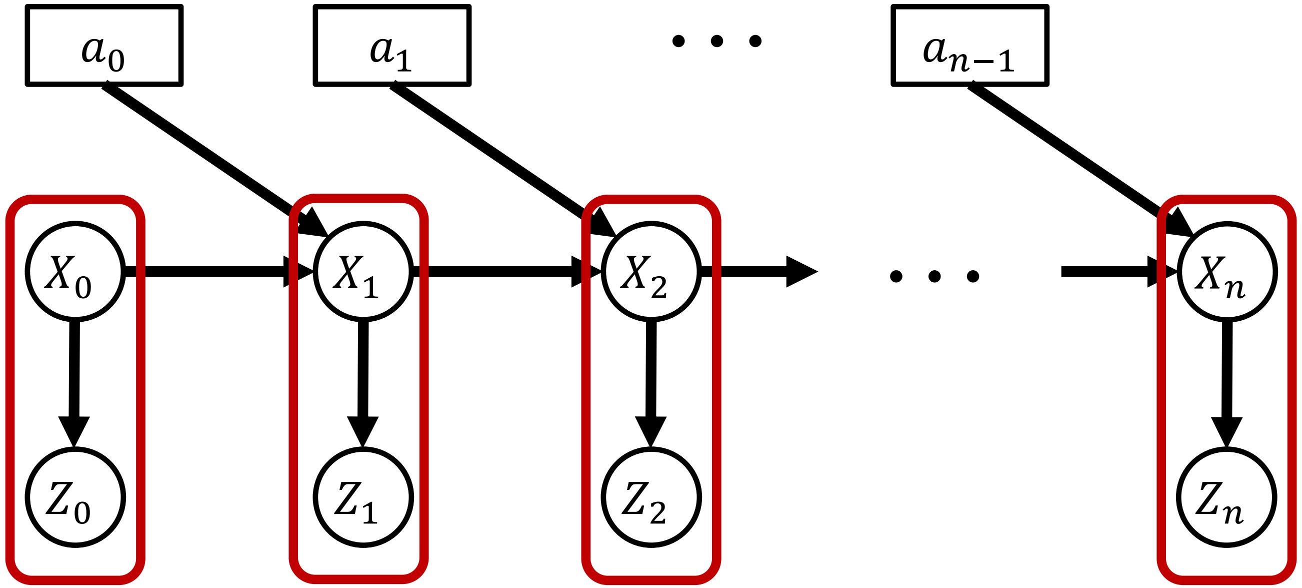
For a controlled HMM with states $X_0, \dots, X_n$, and observations $Z_0, \dots, Z_n$, *the* joint distribution is given by:

$$P(Z_0, \dots, Z_n, X_0 \dots X_n | a_0 \dots a_n) = P(Z_0|X_0)P(X_0) \prod_i P(Z_i|X_i)P(X_i|X_{i-1}, a_i)$$



Simulation Revisited

- Forward simulation is easy for Dynamic Bayes Nets (DBNs).
- Sample initial state x_0 from the prior $P(X_0)$
- For each k
  - generate a sample z_k from the distribution $P(Z_k|X_k = x_k)$
  - generate a sample x_{k+1} from the distribution $P(X_{k+1}|X_k = x_k, a_k)$



Perception

As before, perception is the problem of inferring things about the world given sensor information and context.

For our controlled HMM, we have

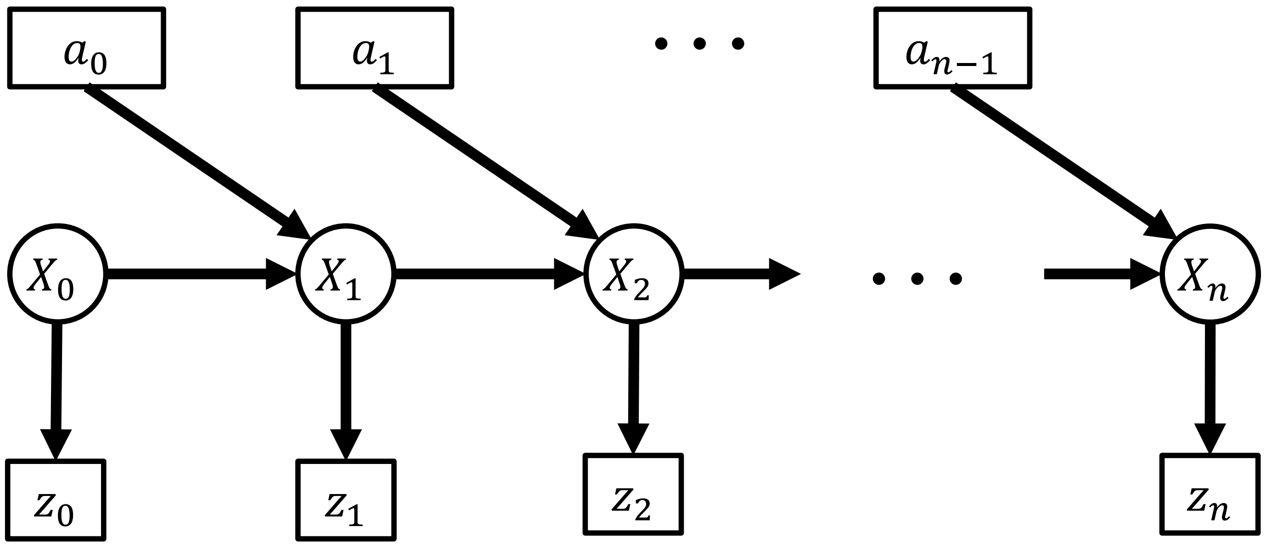
- a sequence of given measurements $Z_t = z_t$
- the known sequence of applied actions $a_1, \dots, a_n$

and we want to infer the states, $X_1, \dots, X_n$

➢ ***There is a lot of structure in this problem, and we can exploit this structure to obtain computationally efficient inference algorithms.***

Hidden Markov Models (HMMs)

- Notice that in the system shown below,
  - we know $\boldsymbol{Z_t = z_t}$ for all $\boldsymbol{t}$
  - We know $\boldsymbol{a_t}$ for all $\boldsymbol{t}$
- We do not know any of $\boldsymbol{X_0 \ldots X_1}$, but we do know that the states form a Markov chain.
- We say that the states, $\boldsymbol{X_0 \ldots X_n}$, are hidden.



HMMs are a good model for speech recognition systems:

- Spoken words behave like a Markov chain (if you know the current word, you know a lot about what will be the next word).
- Measurements are audio signals.

Note: If we increase the relevant history, e.g., so that state X_t depends on $X_{t-1}, X_{t-2} \ldots X_{t-n}$, we have an nth order Markov chain.
Larger n gives better prediction.

Inference in Bayes Nets

Our perception problem is straightforward:

- Given $\mathrm{Z}_1 = z_1 \ldots Z_n = z_n$, and the sequence of applied actions $a_1, \ldots, a_n$,
- Infer the states, $X_1, \ldots, X_n$

The description of the problem almost immediately tells us the mathematical specification:

- ➢ Use $\mathrm{P}(X_1, \ldots, X_n \mid Z_1 = z_1 \ldots Z_n = z_n, a_1, \ldots, a_n)$ to determine an estimate of the state sequence.

Most Probable Explanation

- Recall the definition of conditional probability:

$$\boldsymbol{P(A,B) = P(A|B)P(B)}$$

- We want to compute $\boldsymbol{P(X|Z,A)}$:

$$\boldsymbol{P(X|Z,A) = \frac{P(X,Z,A)}{P(Z,A)} \propto P(X,Z,A)}$$

- We know how to compute $\boldsymbol{P(X,Z,A)}$! (Bayes net magic)

$$\boldsymbol{X = X_1, \dots X_n}$$
$$\boldsymbol{Z = Z_1, \dots Z_n}$$
$$\boldsymbol{A = a_1, \dots a_n}$$

Most Probable Explanation

We are given $Z_t = z_t$, and a_t for all t.

For every possible value of $x_0, \dots, x_n$, compute

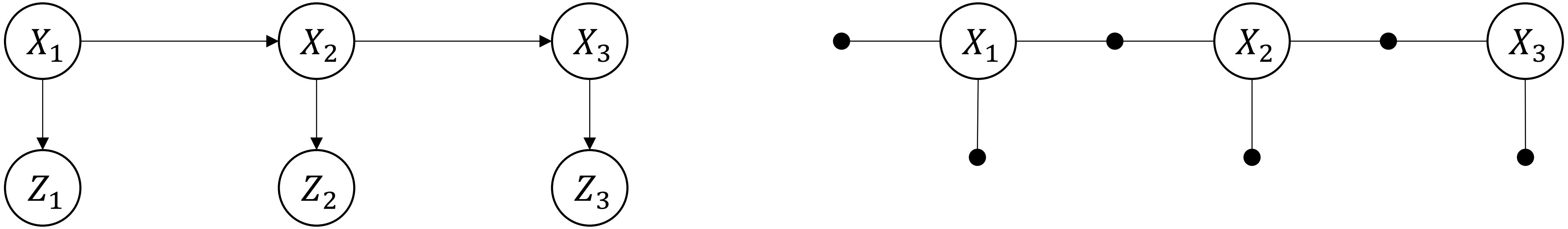
$$P(X, Z, A) = P(Z_0 = z_0 | X_0 = x_0) P(X_0 = x_0) \prod_i P(Z_i = z_i | X_i = x_i) P(X_i = x_i | X_{i-1} = x_{i-1}, a_i)$$

Our estimate is given by

$$X^* = arg\ max_X\ P(X, Z, A)$$

Not the most efficient algorithm, but in principle, this gets the job done.

Factor Graphs



- Measurements are given – get rid of them!

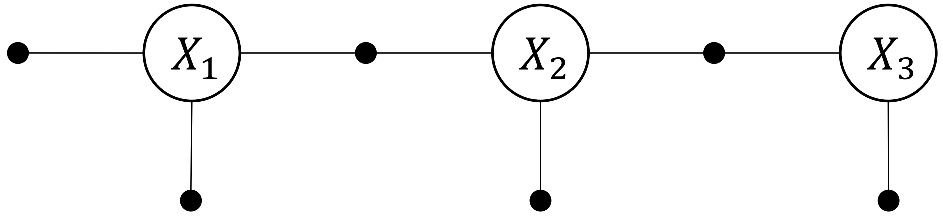
$$P(\mathcal{X}|\mathcal{Z}) \propto P(X_1)L(X_1;z_1)P(X_2|X_1)L(X_2;z_2)P(X_3|X_2)L(X_3;z_3)$$

- This becomes:

$$\phi(\mathcal{X}) = \phi_1(X_1)\phi_2(X_1)\phi_3(X_1,X_2)\phi_4(X_2)\phi_5(X_2,X_3)\phi_6(X_3)$$

Each factor defines a function ϕ which is a function only of its (non-factor node) neighbors.

General definition of Factor graphs



- Bipartite graph of variables and factors

$$\phi(\mathcal{X}) = \prod_i \phi_i(\mathcal{X}_i)$$

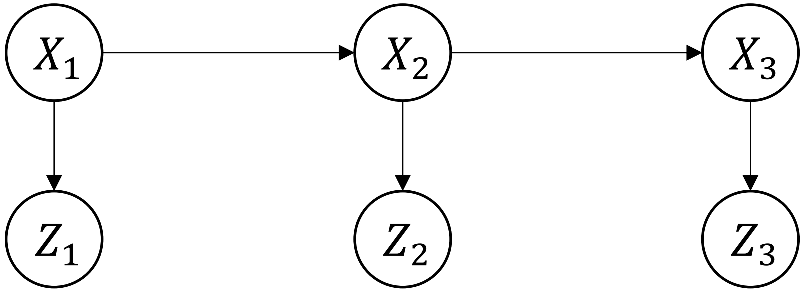
- Each $\mathcal{X}_i$ is the subset of variables connected to factor ϕ_i

Subsets here are:

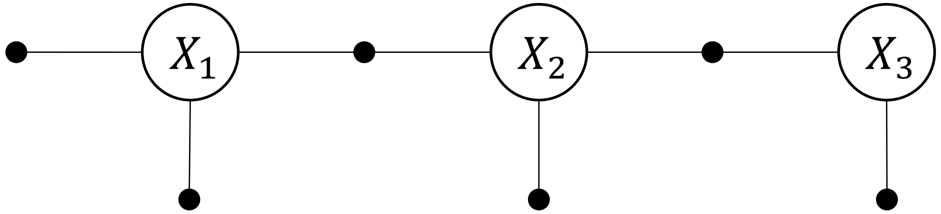
$$\mathcal{X}_1 = \{X_1\}$$
$$\mathcal{X}_2 = \{X_1\}$$
$$\mathcal{X}_3 = \{X_1, X_2\}$$
$$\mathcal{X}_4 = \{X_2\}$$
$$\mathcal{X}_5 = \{X_2, X_3\}$$
$$\mathcal{X}_6 = \{X_3\}$$

Example

$$P(\mathcal{X}|\mathcal{Z}) \propto P(X_1)L(X_1; z_1)P(X_2|X_1)L(X_2; z_2)P(X_3|X_2)L(X_3; z_3)$$



$$\phi(\mathcal{X}) = \phi_1(X_1)\phi_2(X_1)\phi_3(X_1, X_2)\phi_4(X_2)\phi_5(X_2, X_3)\phi_6(X_3)$$



$$\phi_6(X_3) \doteq L(X_3; z_3).$$