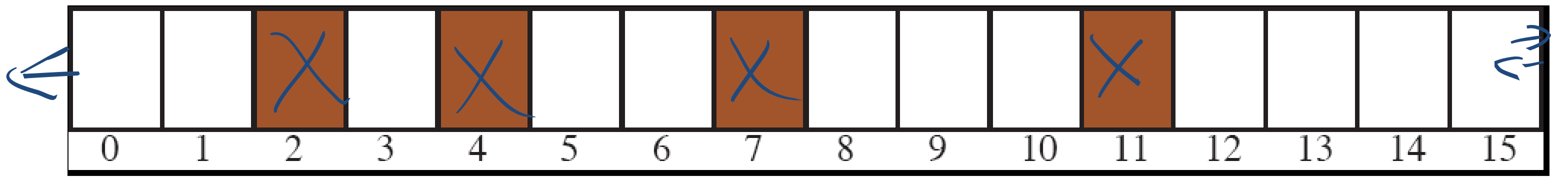
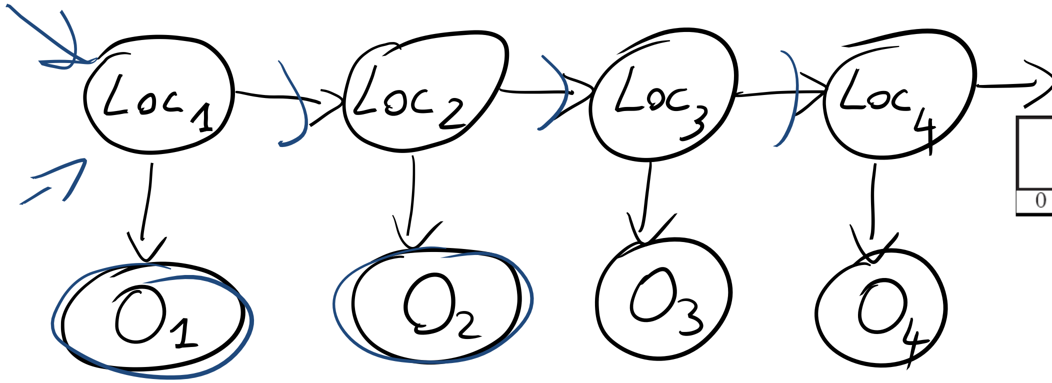
Example: Localization for "Pushed around" Robot

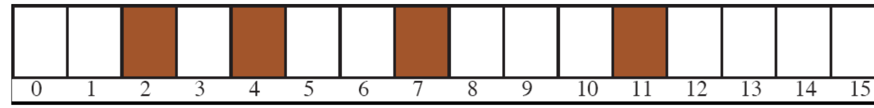
- **Localization** (where am I?) is a fundamental problem in **robotics**
- Suppose a robot is in a circular corridor with 16 locations



- There are four doors at positions: 2, 4, 7, 11
- The Robot initially doesn't know where it is
- The Robot is pushed around. After a push it can stay in the same location, move left or right.
- The Robot has a Noisy sensor telling whether it is in front of a door

This scenario can be represented as…





- Example Stochastic Dynamics: when pushed, it stays in the same location p=0.2, moves one step left or right with equal probability

$P(Loc_{t+1} \,/\, Loc_t)$

$Loc_t = 10$

A.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	.2	0	.4	0	.2	0	0	0

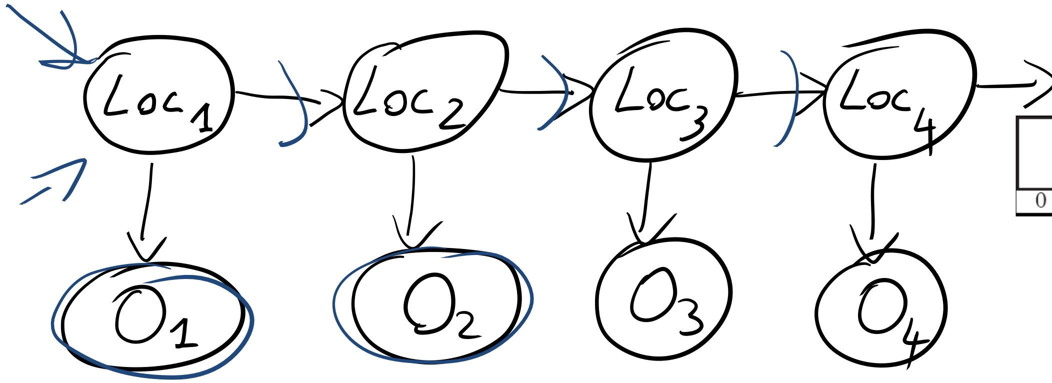
B.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	0	.4	.2	.4	0	0	0	0

C.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0	.2	.2	.2	.2	.2	0	0	0

This scenario can be represented as…



- Example Stochastic Dynamics: when pushed, it stays in the same .4 location p=0.2, moves left or right with equal probability

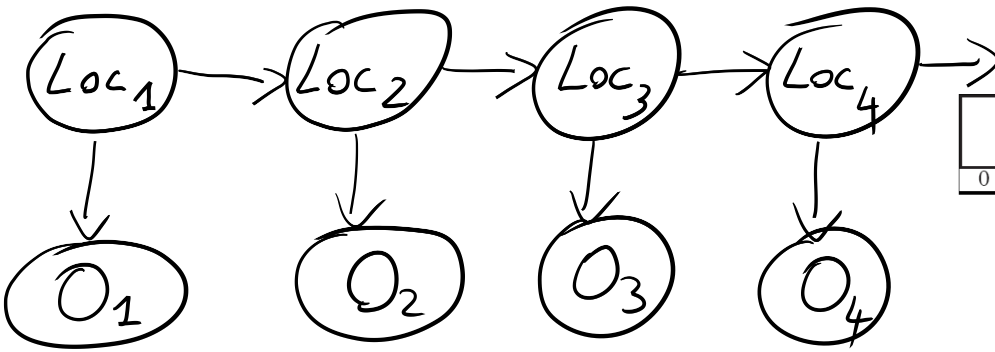
$P(Loc_{t+1} / Loc_t)$

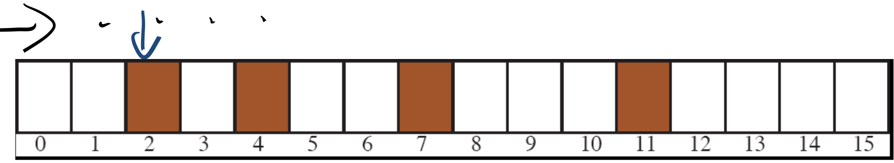
Loc_t \ Loc_{t+1}	0	1	2	….	15
0	.2	.4	0	…. 0	.4
1	.4	.2	.4	0 ….	0
2					
3					
⋮					
15					

$P(Loc_1) =$

0	1	2	…	15
1/16	1/16	1/16	…	

This scenario can be represented as…





$dom(Loc_i) = \{0, 1, \ldots, 15\}$

Example of Noisy sensor telling whether it is in front of a door.

$\boxed{P(O_t / Loc_t)}$

- If it is in front of a door $P(O_t = T) = .8$ WRONG! $\boxed{.2}$
- If not in front of a door $P(O_t = T) = \boxed{.1}$ .9 → 0

Loc_t	$P(O_t=T)$	$P(O_t=F)$
0	.1	.9
1	.1	.9
2	.8	.2
3	.1	.9
4	.8	.2
⋮		

16 prob. distributions

Useful inference in HMMs

- Localization: Robot starts at an unknown location and it is pushed around t times. It wants to determine where it is

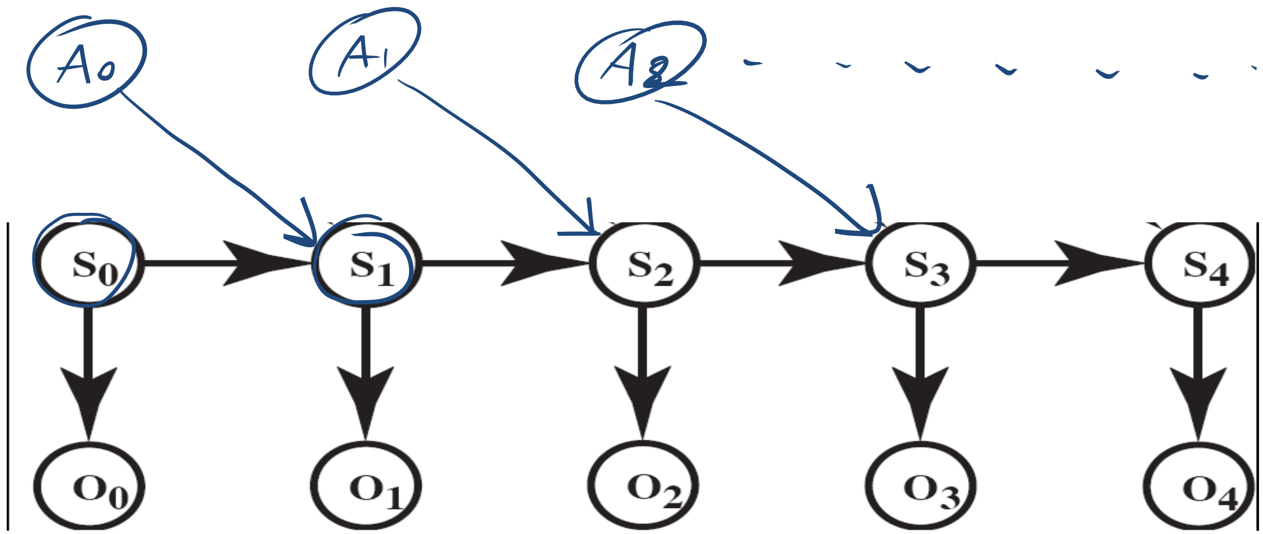
$$P(Loc_t \mid o_1 \cdots\cdots o_t)$$

- In general: compute the posterior distribution over the current state given all evidence to date

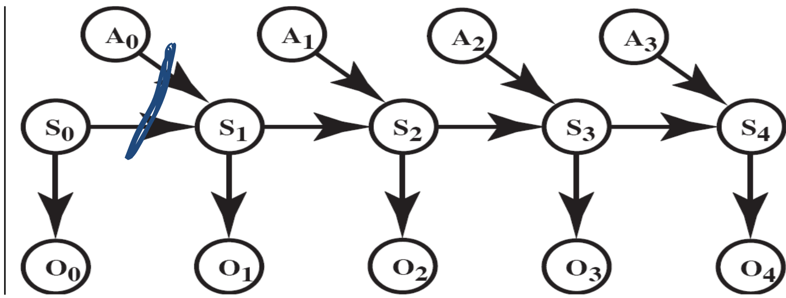
$$P(S_t \mid O_0 \cdots O_t)$$

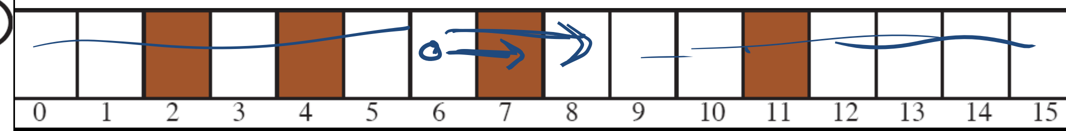
Example : Robot Localization

- Suppose a robot wants to determine its location based on its actions and its sensor readings
- Three actions: *goRight, goLeft, Stay*
- This can be represented by an augmented HMM



Robot Localization Sensor and Dynamics Model





$P(O_t \mid Loc_t)$

- Sample Sensor Model (assume same as for pushed around)
- Sample Stochastic Dynamics: $P(Loc_{t+1} \mid Action_t, Loc_t)$

$P(Loc_{t+1} = L \mid Action_t = goRight, Loc_t = L) = 0.1$

$P(Loc_{t+1} = L+1 \mid Action_t = goRight, Loc_t = L) = 0.8$

$P(Loc_{t+1} = L+2 \mid Action_t = goRight, Loc_t = L) = 0.074$

$P(Loc_{t+1} = L' \mid Action_t = goRight, Loc_t = L) = \boxed{0.002}$ *for all other locations L'*

$\times 13$

- All location arithmetic is modulo 16
- The action *goLeft* works the same but to the left

Dynamics Model More Details

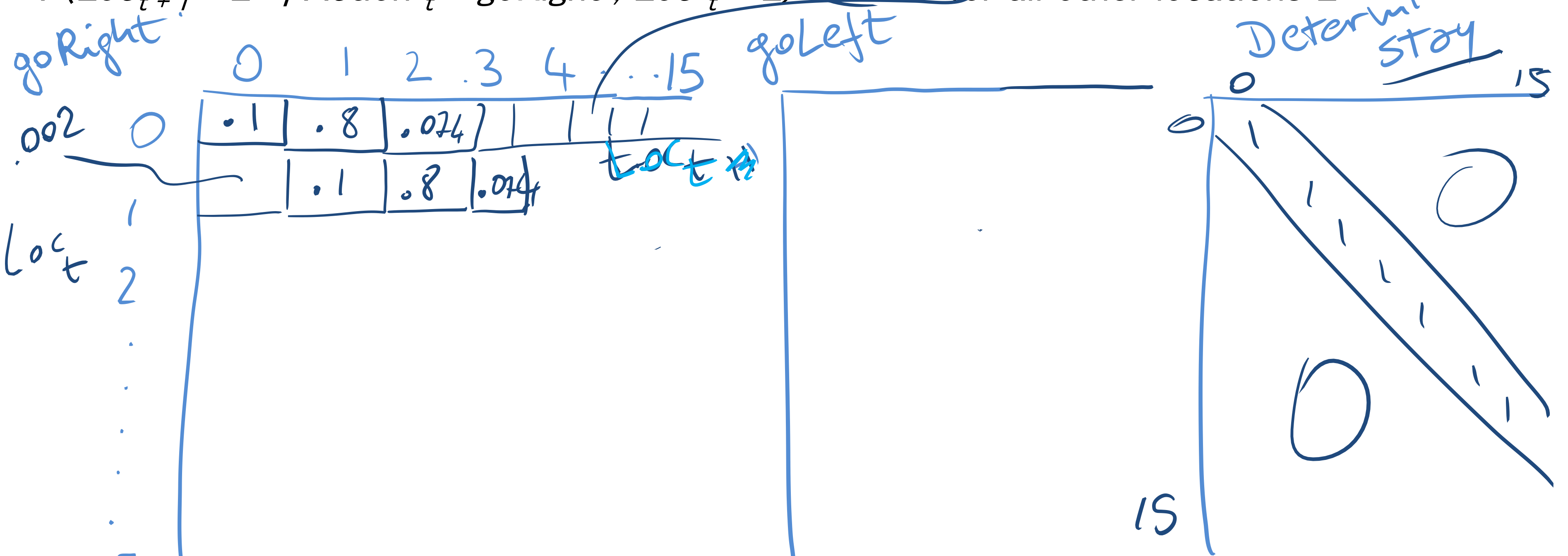
- Sample Stochastic Dynamics: $P(Loc_{t+1} \,/\, Action, Loc_t)$

$P(Loc_{t+1} = L \,/\, Action_t = goRight, Loc_t = L) = \boxed{0.1}$

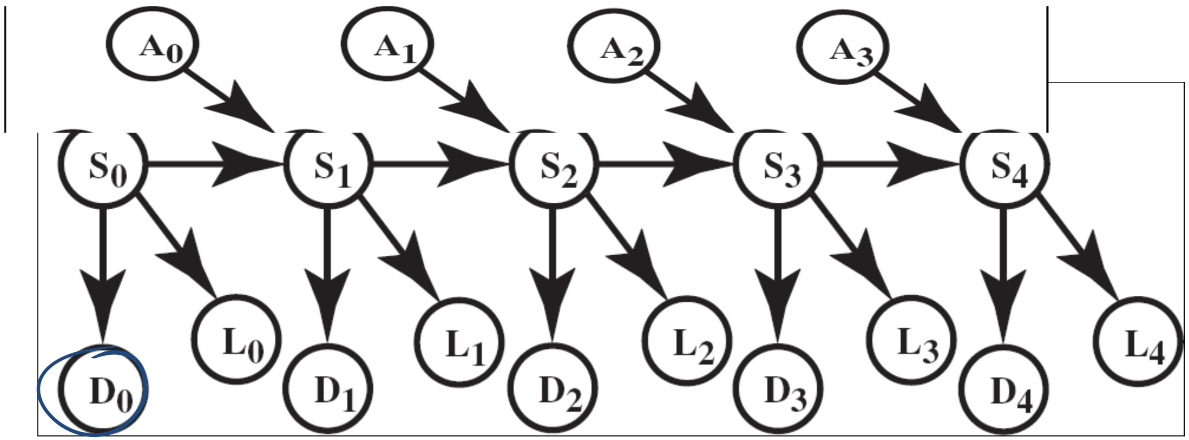
$P(Loc_{t+1} = L+1 \,/\, Action_t = goRight, Loc_t = L) = 0.8$

$P(Loc_{t+1} = L + 2 \,/\, Action_t = goRight, Loc_t = L) = 0.074$

$P(Loc_{t+1} = L' \,/\, Action_t = goRight, Loc_t = L) = 0.002$ *for all other locations L'*



Robot Localization additional sensor



$L_t = T$

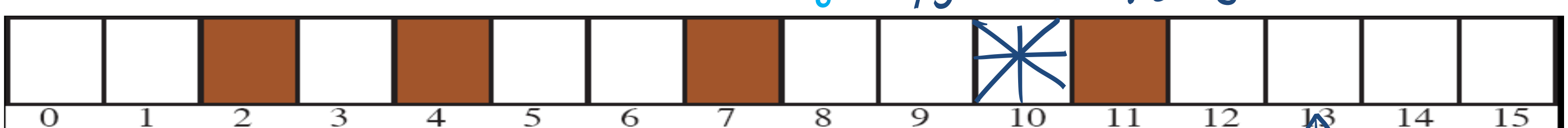
the Robot senses light

- Additional Light Sensor: there is light coming through an opening at location 10

$P(L_t / Loc_t)$

$P(L_t=F)$ .2 .05 .01 .05 .2 .4

$P(L_t=T)$ .8 .95 .99 .95 .8 .6



- Info from the two sensors is combined :“Sensor Fusion”

The Robot starts at an unknown location and must determine where it is

The model appears to be too ambiguous

- Sensors are too noisy
- Dynamics are too stochastic to infer anything

But inference actually works pretty well.

You can check it at :

`http://www.cs.ubc.ca/spider/poole/demos/localization/localization.html`

You can use standard Bnet inference. However you typically take advantage of the fact that time moves forward (not in 322)

Sample scenario to explore in demo

- Keep making observations without moving. What happens?
- Then keep moving without making observations. What happens?
- Assume you are at a certain position alternate moves and observations
- ….

Decision Theory: Single Stage Decisions

Computer Science cpsc322, Lecture 33

(Textbook Chpt 9.2)

June 22, 2017

Lecture Overview

- Intro
- One-Off Decision Example
- Utilities / Preferences and optimal Decision
- Single stage Decision Networks

Planning in Stochastic Environments

Environment

Problem		Deterministic	Stochastic
Static	Constraint Satisfaction	*Vars + Constraints* Arc Consistency Search → for CSP SLS	
	Query	*Logics* CSP Search → for Inference	*Belief Nets* Var. Elimination *Markov Chains and HMMs*
Sequential	Planning	*STRIPS* → CSP Search → for complex planning	*Decision Nets* Var. Elimination

Representation

Reasoning Technique

Planning Under Uncertainty: Intro

- **Planning** how to select and organize a sequence of actions/decisions to achieve a given goal.
- **Deterministic Goal**: A possible world in which some propositions are true

- **Planning under Uncertainty**: how to select and organize a sequence of actions/decisions to "*maximize the probability*" of "*achieving a given goal*"
- **Goal under Uncertainty**: we'll move from all-or-nothing goals to a richer notion: rating how *happy* the agent is in different possible worlds.

"Single" Action vs. Sequence of Actions

Set of primitive decisions that can be treated as a single macro decision to be made *before acting*

one-off

- Agents makes observations
- Decides on an action
- Carries out the action

Sequential Decisions

Lecture Overview

- Intro
- One-Off Decision Example
- Utilities / Preferences and Optimal Decision
- Single stage Decision Networks

One-off decision example

Delivery Robot Example

not

A true

A false

- Robot needs to reach a certain room
- Going through stairs may cause an accident.
- It can go the short way through long stairs, or the long way through short stairs (that reduces the chance of an accident but takes more time)

which way long

short

$P(A=t \mid WW=long) < P(A=t \mid WW=short)$

- The Robot can choose to wear pads to protect itself or not (to protect itself in case of an accident) but pads slow it down

Wear Pads t

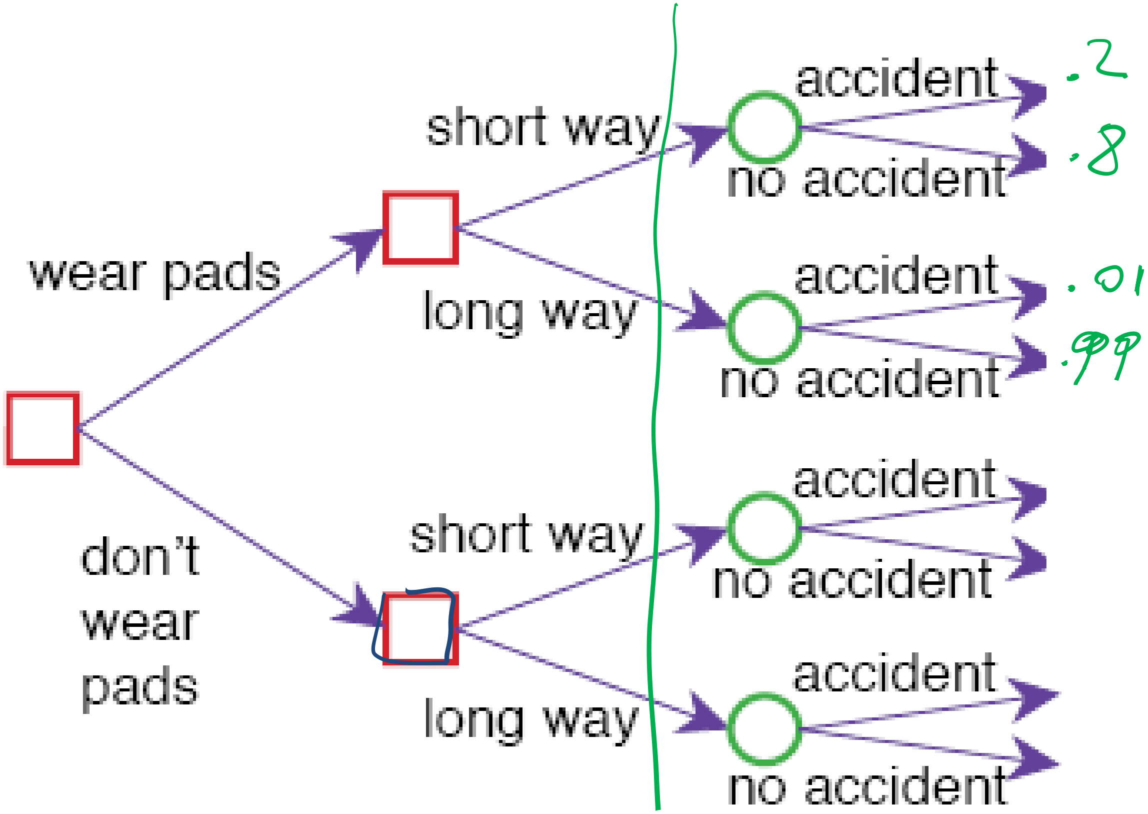
f

if $A=t$

- If there is an accident the Robot does not get to the room

Decision Tree for Delivery Robot

- This scenario can be represented as the following decision tree

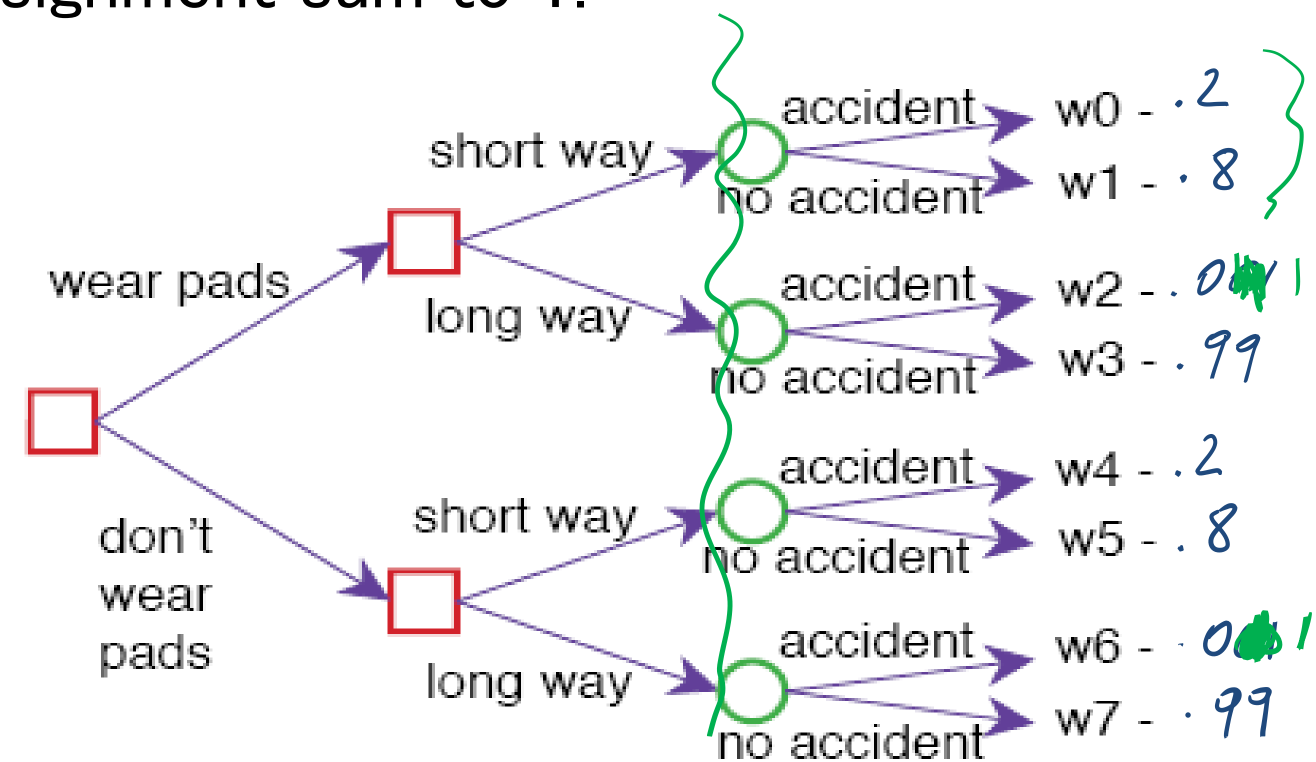


Which way	Accident	
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

- The agent has a set of decisions to make (a macro-action it can perform)
- Decisions can influence random variables
- Decisions have probability distributions over outcomes

Decision Variables: Some general Considerations

- A possible world specifies a value for each random variable and each decision variable.
- For each assignment of values to all decision variables, the probabilities of the worlds satisfying that assignment sum to 1.



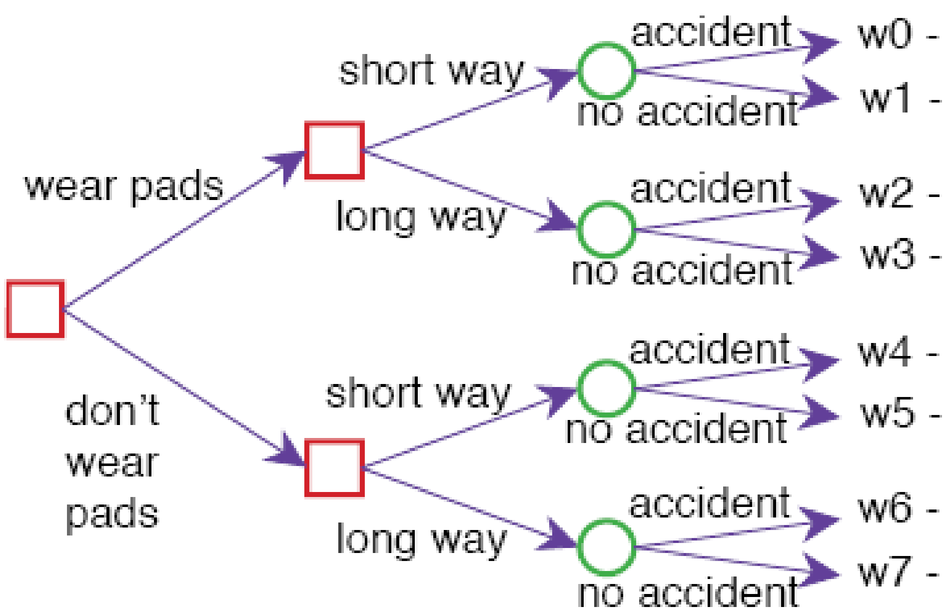
Lecture Overview

- Intro
- One-Off Decision Problems
- Utilities / Preferences and Optimal Decision
- Single stage Decision Networks

What are the optimal decisions for our Robot?

It all depends on how happy the agent is in different situations.

For sure getting to the room is better than not getting there….. but we need to consider other factors..



Utility / Preferences

Utility: a measure of desirability of possible worlds to an agent

- Let U be a real-valued function such that $U(w)$ represents an agent's degree of preference for world w . [0, 100]

Would this be a reasonable utility function for our Robot, who wants to reach the room?

i▸clicker.

Which way	Accident	Wear Pads	Utility	World
short	true	true	35	w0, moderate damage
short	false	true	95	w1, reaches room, quick, extra weight
long	true	true	30	w2, moderate damage, low energy
long	false	true	75	w3, reaches room, slow, extra weight
short	true	false	3	w4, severe damage
short	false	false	100	w5, reaches room, quick
long	false	false	0	w6, reaches room, slow
long	true	false	80	w7, severe damage, low energy

A. Yes

C . No

B. It depends

Utility: Simple Goals

- How can the simple (boolean) goal "reach the room" be specified?

i>clicker.

A.

Which way	Accident	Wear Pads	Utility
long	true	true	0
long	true	false	0
long	false	true	0
long	false	false	0
short	true	true	0
short	true	false	0
short	false	true	100
short	false	false	90

B.

Which way	Accident	Wear Pads	Utility
long	true	true	0
long	true	false	0
long	false	true	0
long	false	false	100
short	true	true	0
short	true	false	0
short	false	true	0
short	false	false	0

C.

Which way	Accident	Wear Pads	Utility
long	true	true	0
long	true	false	0
long	false	true	100
long	false	false	100
short	true	true	0
short	true	false	0
short	false	true	100
short	false	false	100

D. ***Not possible***

Utility: Simple Goals

- Can simple (boolean) goals still be specified?

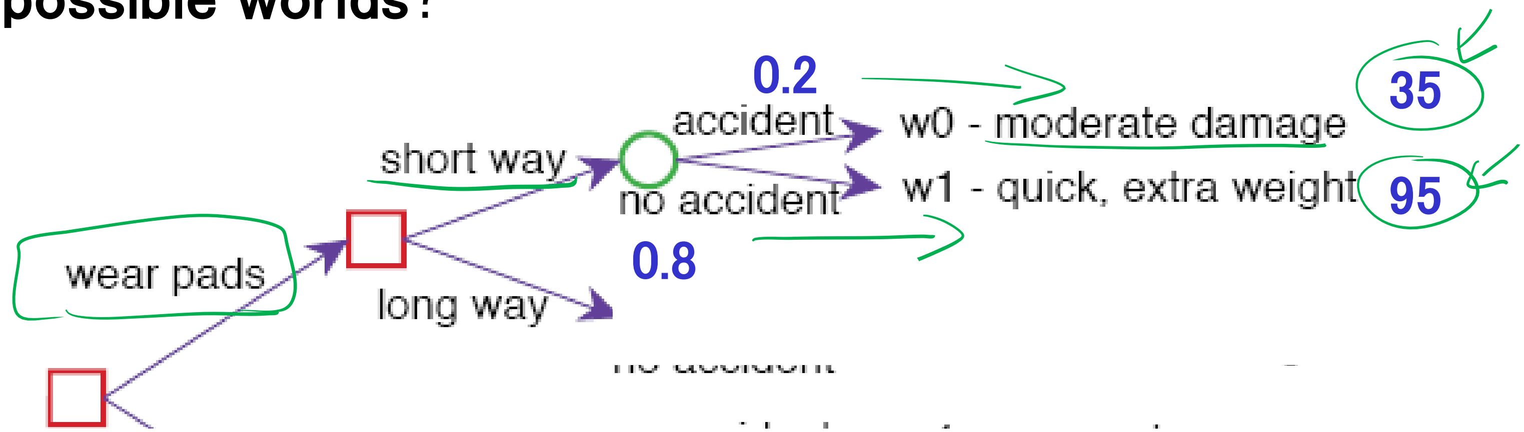
goal : "reaching the room"

Accident must be false

Which way	Accident	Wear Pads	Utility
long	true	true	0
long	true	false	0
long	false	true	100
long	false	false	100
short	true	true	0
short	true	false	0
short	false	true	100
short	false	false	100

Optimal decisions: How to combine Utility with Probability

What is the **utility** of achieving a certain **probability distribution** over **possible worlds**?



- It is its **expected utility/value** i.e., its average utility, weighting possible worlds by their probability.

$EU(wp=t, ww=short) =$

$.2 * 35 + .8 * 95$

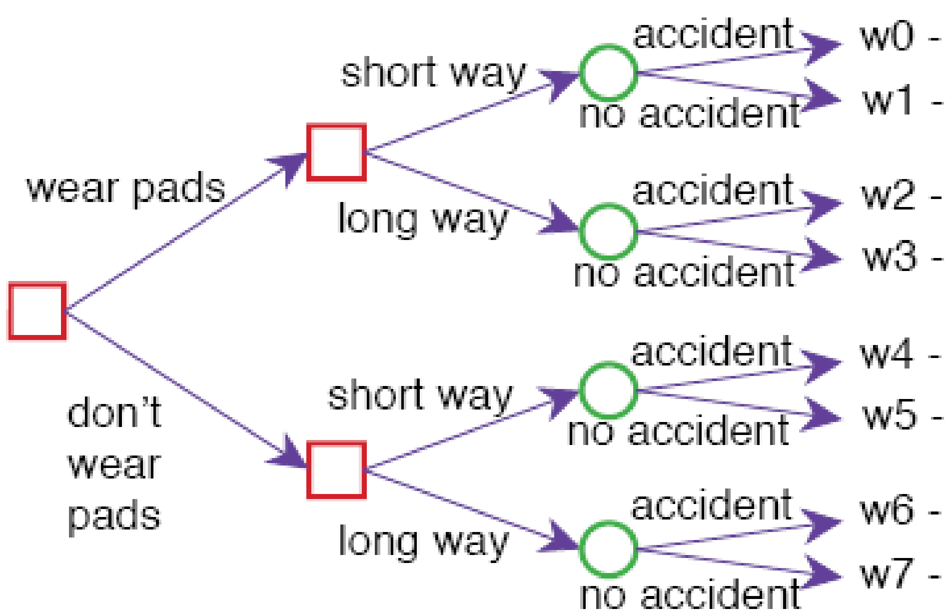
Optimal decision in one-off decisions

- Given a set of n decision variables var_i (e.g., Wear Pads, Which Way), the agent can choose:

  $D = d_i$ for any $d_i \in \text{dom}(var_1) \times .. \times \text{dom}(var_n)$.

d_i

Wear Pads	Which way
true	short
true	long
false	short
false	long

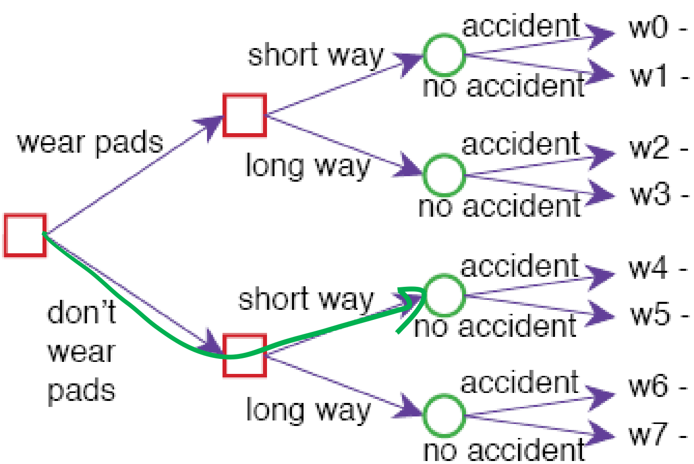


Optimal decision: Maximize Expected Utility

- The expected utility of decision $D = d_i$ is

$$\mathbb{E}(U \mid D = d_i) = \sum_{w \models D = d_i} P(w \mid D = d_i)\, U(w)$$

e.g., $\mathbb{E}(U \mid D = \{WP=$ false $, WW=$ short $\}) =$



$P(w_4 \mid D) * U(w_4) +$
$P(w_5 \mid D) * U(w_5)$

max

- An optimal decision is the decision $D = d_{max}$ whose expected utility is maximal:

$$d_{max} = \underset{d_i \in dom(D)}{\arg\max}\, \mathbb{E}(U \mid D = d_i)$$

Wear Pads	Which way	EU
true	short	
true	long	
false	short	
false	long	

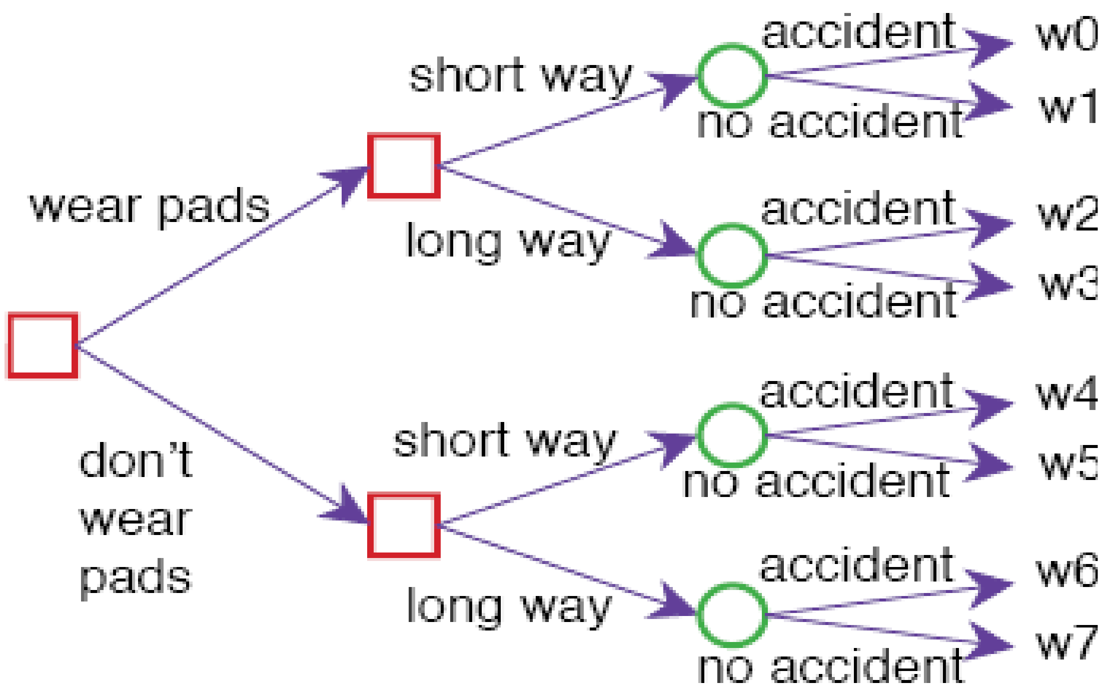
Expected utility of a decision

- The expected utility of decision $D = d_i$ is

$$\mathbb{E}(U \mid D = d_i) = \sum_{w \models (D = d_i)} P(w)\, U(w)$$

- What is the expected utility of Wearpads=true, WhichWay=short ?

  0.2 * 35 + 0.8 * 95 = 83



	Conditional probability	Utility	E[U|D]
w0	0.2	35	83
w1	0.8	95	
w2	0.01	30	74.55
w3	0.99	75	
w4	0.2	3	80.6
w5	0.8	100	
w6	0.01	0	79.2
w7	0.99	80	

Lecture Overview

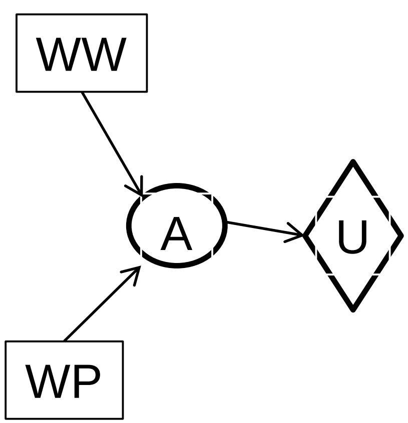
- Intro
- One-Off Decision Problems
- Utilities / Preferences and Optimal Decison
- Single stage Decision Networks

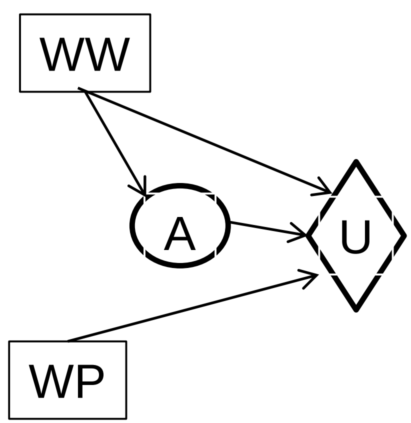
Single-stage decision networks

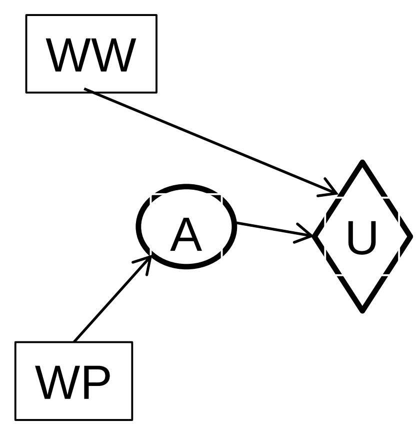
Extend belief networks with:

- **Decision nodes**, that the agent chooses the value for. *Drawn as rectangle*.
- **Utility node**, the parents are the variables on which the utility depends. *Drawn as a diamond*.
- Shows explicitly which decision nodes affect random variables

iclicker.







A.

B.

C .

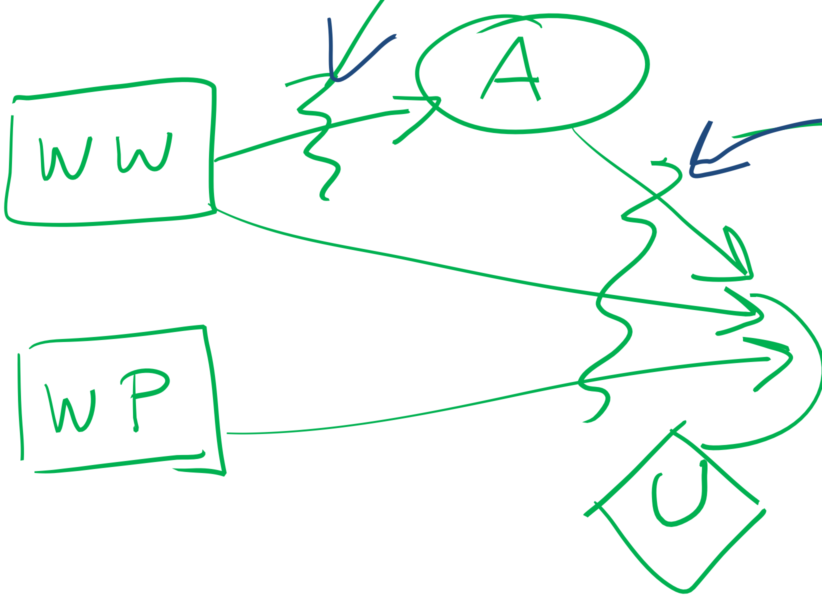
D . None of these

Single-stage decision networks

Extend belief networks with:

- **Decision nodes**, that the agent chooses the value for. *Drawn as rectangle*.
- **Utility node**, the parents are the variables on which the utility depends. *Drawn as a diamond*.
- Shows explicitly which decision nodes affect random variables

Which way	Accident	
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8



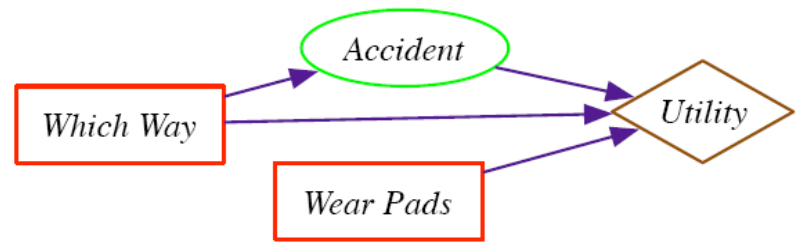
Which way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Finding the optimal decision: We can use VE

Suppose the random variables are $X_1, \cdots, X_n$, the decision variables are the set D, and utility depends on

$pU \subseteq \{X_1, \cdots, X_n\} \cup D$

parents U



$$\mathbb{E}(U \mid D) = \sum_{X_1,\dots,X_n} P(X_1,\dots,X_n \mid D)\, U(pU)$$

$$= \sum \prod P(x_i \mid pX_i)\ U(pU)$$

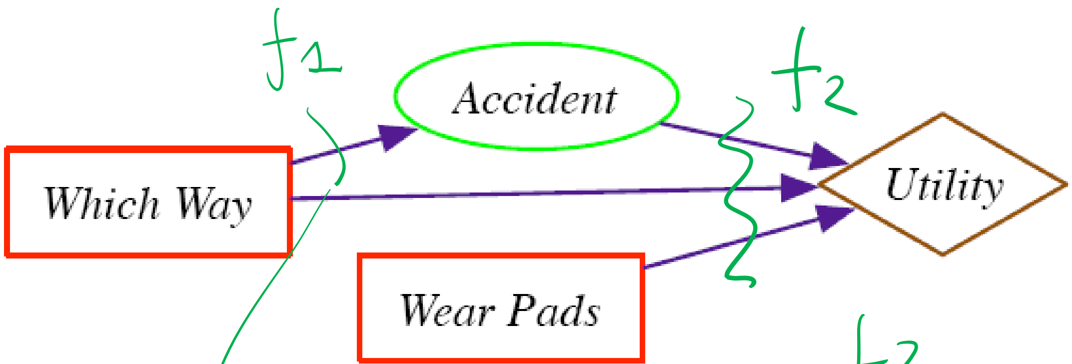
decision vars

also includes

To find the optimal decision we can use VE:

1. Create a factor for each conditional probability and for the utility
2. Multiply factors and sum out all of the random variables (This creates a factor on D that gives the expected utility for each d_1)
3. Choose the d_1 with the maximum value in the factor.

Example Initial Factors (Step1)



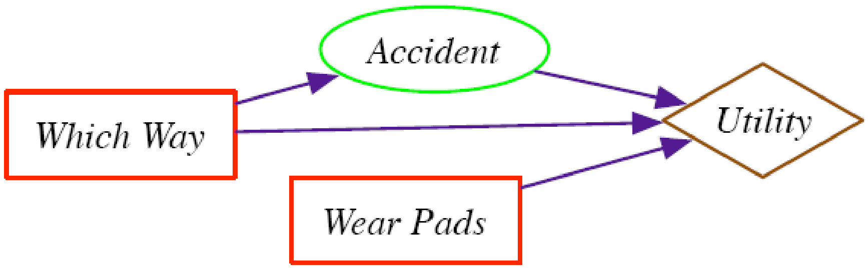
f2

Which way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

f1

Which way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

Example: Multiply Factors (Step 2a)



$$\sum_A \overbrace{f_1(WW, A) \times f_2(A, WW, WP)}^{f}$$

f_3

f_1

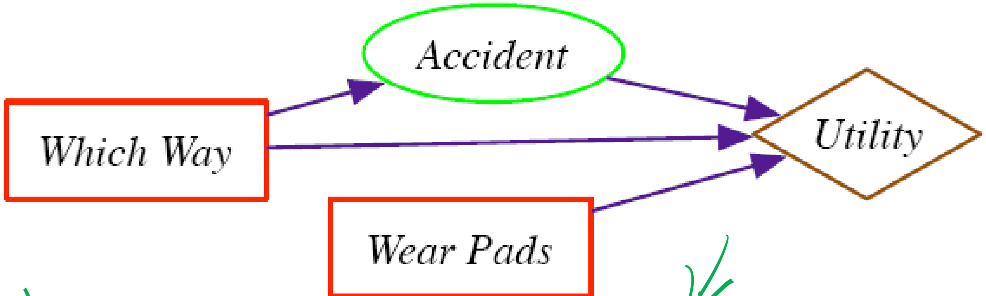
Which way	Accident	Probability
long	true	0.01
long	false	0.99
short	true	0.2
short	false	0.8

f_2 →

Which way	Accident	Wear Pads	Utility
long	true	true	30
long	true	false	0
long	false	true	75
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Which way	Accident	Wear Pads	Utility
long	true	true	30 * .01
long	true	false	0 * .01
long	false	true	75 * .99
long	false	false	80
short	true	true	35
short	true	false	3
short	false	true	95
short	false	false	100

Example: Sum out vars and choose max (Steps 2b–3)



$$\sum_{A} f'(A, WW, WP)$$

Sum out accident:

Which way	Accident	Wear Pads	Utility
long	true	true	0.01*30
long	true	false	0.01*0
long	false	true	0.99*75
long	false	false	0.99*80
short	true	true	0.2*35
short	true	false	0.2*3
short	false	true	0.8*95
short	false	false	0.8*100

Which way	Wear Pads	Expected Utility
long	true	0.01*30+0.99*75=74.55
long	false	0.01*0+0.99*80=79.2
short	true	0.2*35+0.8*95=83
short	false	0.2*3+0.8*100=80.6

Thus the optimal policy is to take the **short way** and **wear pads**, with an ***expected utility*** of 83.

Learning Goals for today's class

You can:

- Compare and contrast stochastic **single-stage (one-off)** decisions vs. **multistage** decisions
- Define a **Utility Function** on possible worlds
- Define and compute **optimal one-off decision** (max expected utility)
- Represent one-off decisions as **single stage decision networks** and compute optimal decisions by **Variable Elimination**

Next Class (textbook sec. 9.3)

Set of primitive decisions that can be treated as a single macro decision to be made *before acting*

Sequential Decisions

- Agents makes observations
- Decides on an action
- Carries out the action