Lecture 23

Planning Under Uncertainty and Decision Networks

Lecture Overview

- Recap
- Intro to Decision theory
- Utility and expected utility
- Decision Networks for Single-stage decision problems

Inference in General

- Y: subset of variables that is queried (e.g. Temperature in previous example)
- E: subset of variables that are observed . E = e (W = yes in previous example)
- $Z_1, \ldots, Z_k$ remaining variables in the JPD (Cloudy in previous example)

We need to compute this numerator for each value of Y, y_i

We need to marginalize over all the variables $Z_1, \ldots Z_k$ not involved in the query

$$P(Y = y_i, E = e) = \sum_{Z_1} \ldots \sum_{Z_k} P(Z_1, \ldots, Z_k, Y = y_i, E = e)$$

$$P(Y \mid E = e) = \frac{P(Y, E = e)}{P(E = e)}$$

Def of conditional probability

$\underline{P(Y,E=e)}$

To compute the denominator, marginalize over Y

$\sum_Y P(Y, E=e)$ constant - Same value for every ensuring that P(Y=$\sum_Y$ y$P(Y$i).

Normalization $= y_i \mid E) = 1$

- All we need to compute is the numerator: joint probability of the query variable(s) and the evidence!
- Variable Elimination is an algorithm that efficiently performs this operation by casting it as operations between factors - introduced next

Factors

- A factor is a function from a tuple of random variables to the real numbers R
- We write a factor on variables $X_1, \ldots, X_j$ as $f(X_1, \ldots, X_j)$

- $P(X_1, X_2)$ is a factor $f(X_1, X_2)$

  Distribution

- $P(Z \mid X,Y)$ is a factor $f(Z,X,Y)$

  Set of Distributions

  One for each combination of values for X and Y

- $P(Z=f \mid X,Y)$ is a factor $f(X,Y)$

  $f(X, Y)_{Z=f}$

  Set of partial Distributions

- Note: Factors do not have to sum to one

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

- A factor denotes one or more (possibly partial) distributions over the given tuple of variables, e.g.,

Recap: Factors and Operations on Them

If we assign variable A=a in factor $f_7(A,B)$, what is the correct form for the resulting factor? • f(B).

When we assign variable A we remove it from the factor's domain

If we marginalize variable A out from factor $f_7(A,B)$, what is the correct form for the resulting factor? • f(B).

When we marginalize out variable A we remove it from the factor's domain

If we multiply factors $f_4(X,Y)$ and $f_6(Z,Y)$, what is the correct form for the resulting factor?

- f(X,Y,Z)
- When multiplying factors, the resulting factor's domain is the union of the multiplicands' domains
- What is the correct form for $\sum_B f_5(A,B) \times f_6(B,C)$

- As usual, product before sum: $\sum_B (\; f_5(A,B) \times f_6(B,C) \;)$
- Result of multiplication: $f_7(A,B,C)$. Then marginalize out B: $f_8(A,C)$

The variable elimination algorithm,

n

The JPD of a Bayesian network is $P(X_1, \ldots, X_n) = \prod_{i=1} P(X_i \mid pa(X_i))$

Given: $P(Y, E_1, \ldots, E_j, Z_1, \ldots, Z_k)$ observed Other variables not involved in the query

To compute $P(Y=y_i \mid E_1=e_1, \ldots, E_j=e_j) = \dfrac{P(Y=y_i, E_1=e_1, \ldots, E_j=e_j)}{\sum_{Y} P(Y=y, E_1=e_1, \ldots, E_j=e_j)}$

The variable elimination algorithm,

The JPD of a Bayesian network is $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid pa(X_i))$

Given: $P(Y, E_1, \ldots, E_j, Z_1, \ldots, Z_k)$ observed Other variables not involved in the query

To compute $P(Y=y_i \mid E_1=e_1, \ldots, E_j=e_j) = \dfrac{P(Y=y_i, E_1=e_1, \ldots, E_j=e_j)}{\sum_{Y=y} P(Y=y, E_1=e_1, \ldots, E_j=e_j)}$

1. Construct a factor for each conditional probability. $P(X_i \mid pa(X_i)) = f_i(X_i, pa(X_i))$

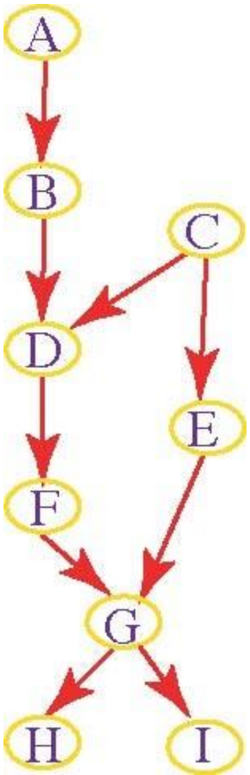
$$P(Y, E_1 = e_1, \ldots, E_j = e_j) = \sum_{Z_1} \cdots \sum_{Z_k} \prod_{i=1}^{n} (f_i)_{E_1=e_1,\ldots,E_j=e_j}$$

2. For each factor, **assign** the observed variables E to their observed values.
3. Given an elimination ordering, decompose sum of products
4. **Sum out** all variables Z_i not involved in the query (one a time)
   - Multiply factors containing Z_i
   - Then marginalize out Z_i from the product
5. Multiply the remaining factors (which only involve Y)
6. **Normalize** by dividing the resulting factor f(Y) by $\sum_y f(Y)$

Step 1: Construct a factor for each cond. probability

Compute $P(G \mid H{=}h_1)$.

$$P(G,H) = \sum_{A,B,C,D,E,F,I} P(A)P(B|A)P(C)P(D|B,C)P(E|C)P(F|D)P(G|F,E)P(H|G)P(I|G)$$



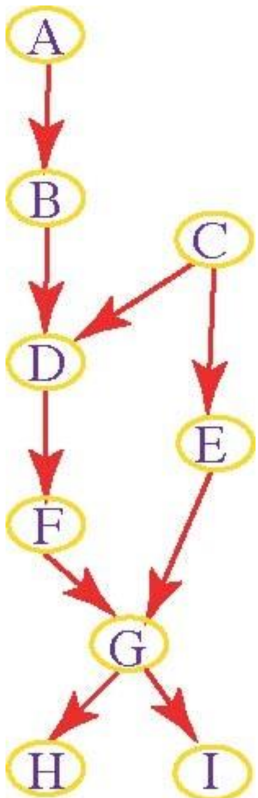
$$P(G,H) = \sum_{A,B,C,D,E,F,I} f_0(A)\ f_1(B,A)\ f_2(C)\ f_3(D,B,C)\ f_4(E,C)\ f_5(F, D)\ f_6(G,F,E)\ f_7(H,G)\ f_8(I,G)$$

Step 2: assign to observed variables their observed values.

Compute $P(G \mid H=h_1)$.

Previous state:

$$P(G,H) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F,D)\, f_6(G,F,E)\, f_7(H,G)\, f_8(I,G)$$



Observe H :

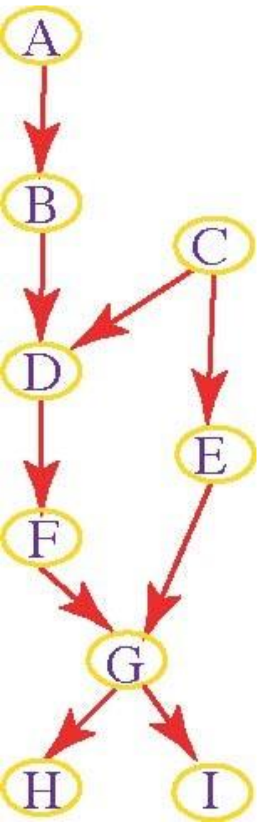
$$P(G,H=h_1)=\sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F,D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$$

Step 3: Decompose sum of products

Compute $P(G \mid H{=}h_1)$.

Previous state:

$$P(G,H{=}h_1) = \sum_{A,B,C,D,E,F,I} f_0(A)\, f_1(B,A)\, f_2(C)\, f_3(D,B,C)\, f_4(E,C)\, f_5(F,D)\, f_6(G,F,E)\, f_9(G)\, f_8(I,G)$$



Elimination ordering A, C, E, I, B, D, F :

$$P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$$

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.

Elimination order: **A**,C,E,I,B,D,F

Previous state:

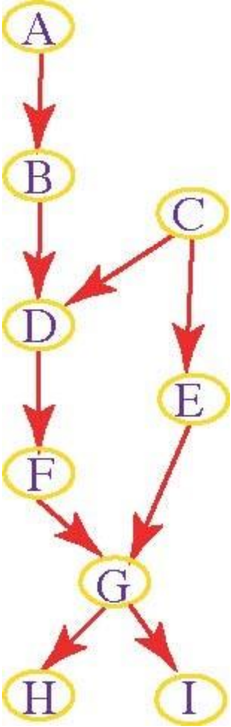
$$P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C) \sum_A f_0(A)\, f_1(B,A)$$

Eliminate A: perform product and sum out *A in*

$$P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$$

Compute

$f_{10}(B)$ does not depend on C, E, or I, so we can push it outside of those sums.



Step 4: sum out non query variables (one at a time)

$P(G \mid H=h_1)$.

Elimination order: A,**C**,E,I,B,D,F

Previous state:

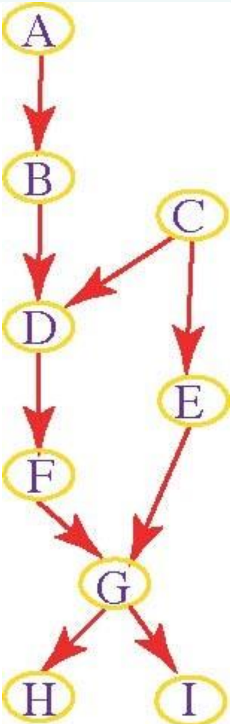
$P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E) \sum_C f_2(C)\, f_3(D,B,C)\, f_4(E,C)$

Eliminate C: perform product and sum out *C in*

$P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(B,D,E)$

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.



Elimination order: A,C,**E**,I,B,D,F

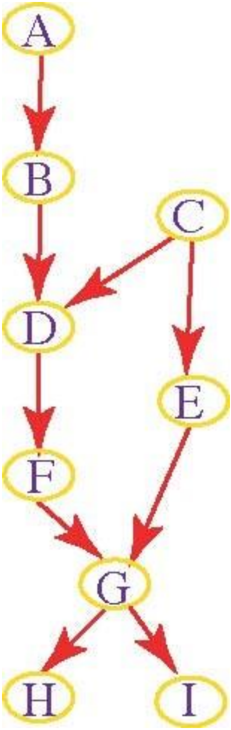
Previous state:

$P(G,H=h_1) = P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B) \sum_I f_8(I,G) \sum_E f_6(G,F,E)\, f_{11}(B,D,E)$

Eliminate E: perform product and sum out *E in*

Compute

$P(G,H=h_1) = P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(B,D,F,G) \sum_I f_8(I,G)$



Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.

Elimination order: A,C,E,**I**,B,D,F

Previous state:

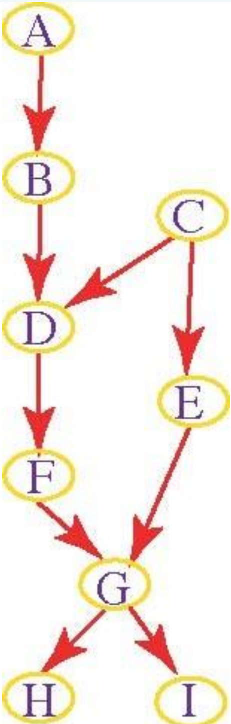
$$P(G,H=h_1) = P(G,H=h_1) = f_9(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(B,D,F,G) \sum_I f_8(I,G)$$

Eliminate I: perform product and sum out *I in*

$$P(G,H=h_1) = P(G,H=h_1) = f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(B,D,F,G)$$

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.



Elimination order: A,C,E,I,**B**,D,F

Previous state:

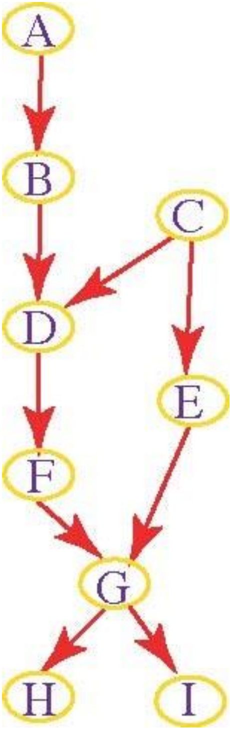
$$P(G,H=h_1) = P(G,H=h_1) = f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D) \sum_B f_{10}(B)\, f_{12}(B,D,F,G)$$

Eliminate B: perform product and sum out *B in*

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H{=}h_1)$.

$P(G,H{=}h_1) = P(G,H{=}h_1) = f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(D,F,G)$



Elimination order: A,C,E,I,B,**D**,F

Previous state:

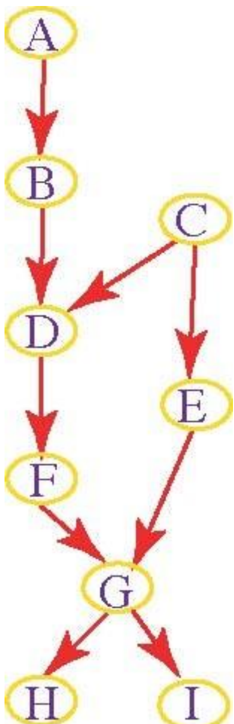
$P(G,H{=}h_1) = P(G,H{=}h_1) = f_9(G)\, f_{13}(G) \sum_F \sum_D f_5(F, D)\, f_{14}(D,F,G)$

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.

Eliminate D: perform product and sum out D *in*

$P(G,H=h_1) = P(G,H=h_1) = f_9(G)\, f_{13}(G) \sum_F f_{15}(F,G)$



Multiply remaining factors (all in G): $P(G,H=h_1) = f_{17}(G)$

Elimination order: A,C,E,I,B,D,F

Previous state:

$P(G,H=h_1) = P(G,H=h_1) = f_9(G)\, f_{13}(G) \sum_F f_{15}(F,G)$

$$P(G=g \mid H=h_1) = \frac{P(G=g, H=h_1)}{P(H=h_1)} = \frac{P(G=g, H=h_1)}{\sum_{g' \in dom(G)} P(G=g', H=h_1)} = \frac{f_{17}(g)}{\sum_{g' \in dom(G)} f_{17}(g')}$$

Step 4: sum out non query variables (one at a time)

Compute $P(G \mid H=h_1)$.

Eliminate F: perform product and sum out F in

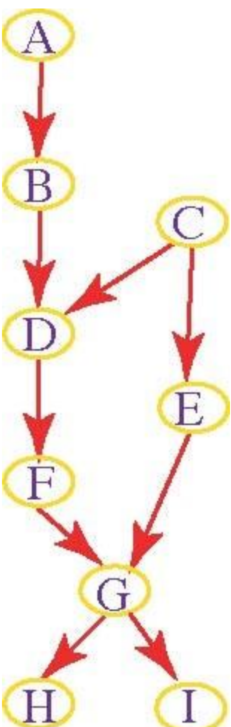
g' $\in dom\ G$

$P(G,H=h_1) = f_9(G)\, f_{13}(G) f_{16}(G)$

Multiply remaining factors (all in G):

$P(G,H=h_1) =$

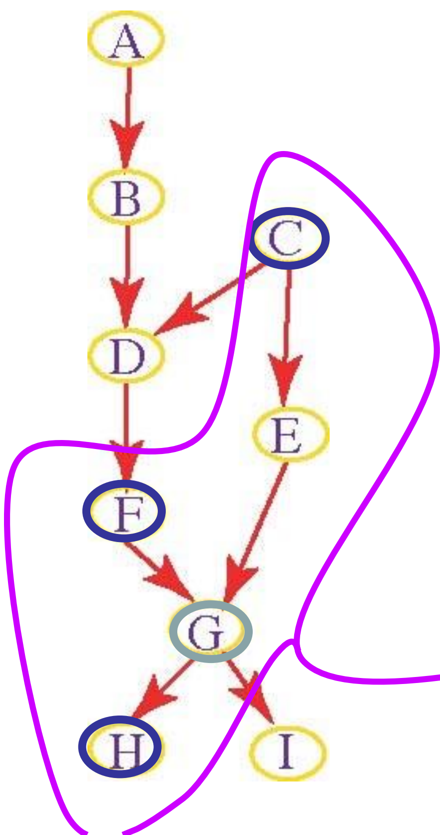
$f_{17}(G)$



Normalize

Variable elimination: pruning

- Before running VE, we can prune all variables Z that are conditionally independent of the query Y given evidence E: $Z \perp\!\!\!\perp Y \mid E$
  - They cannot change the belief over Y given E!



Thus, if the query is
$P(G=g \mid C=c_1, F=f_1, H=h_1)$
we only need to consider this subnetwork

- We can also prune unobserved leaf nodes
  - Since they are unobserved and not predecessors of the query nodes, they cannot influence the posterior probability of the query nodes

Complexity of Variable Elimination (VE) (not required)

- The complexity of VE is exponential in the maximum number of variables in any factor during its execution
- This number is called the treewidth of a graph (along an ordering)

- Elimination ordering influences treewidth
- Finding the best ordering is NP complete
- I.e., the ordering that generates the minimum treewidth
- Heuristics work well in practice (e.g. least connected variables first) • Even with best ordering, inference is sometimes infeasible
  - ✓ In those cases, we need approximate inference. See CS422 & CS540

Lecture Overview

- Recap
- Intro to Decision theory
- Utility and expected utility

- Decision Networks for Single-stage decision problems

Where are we?

Representation

Reasoning

Environment

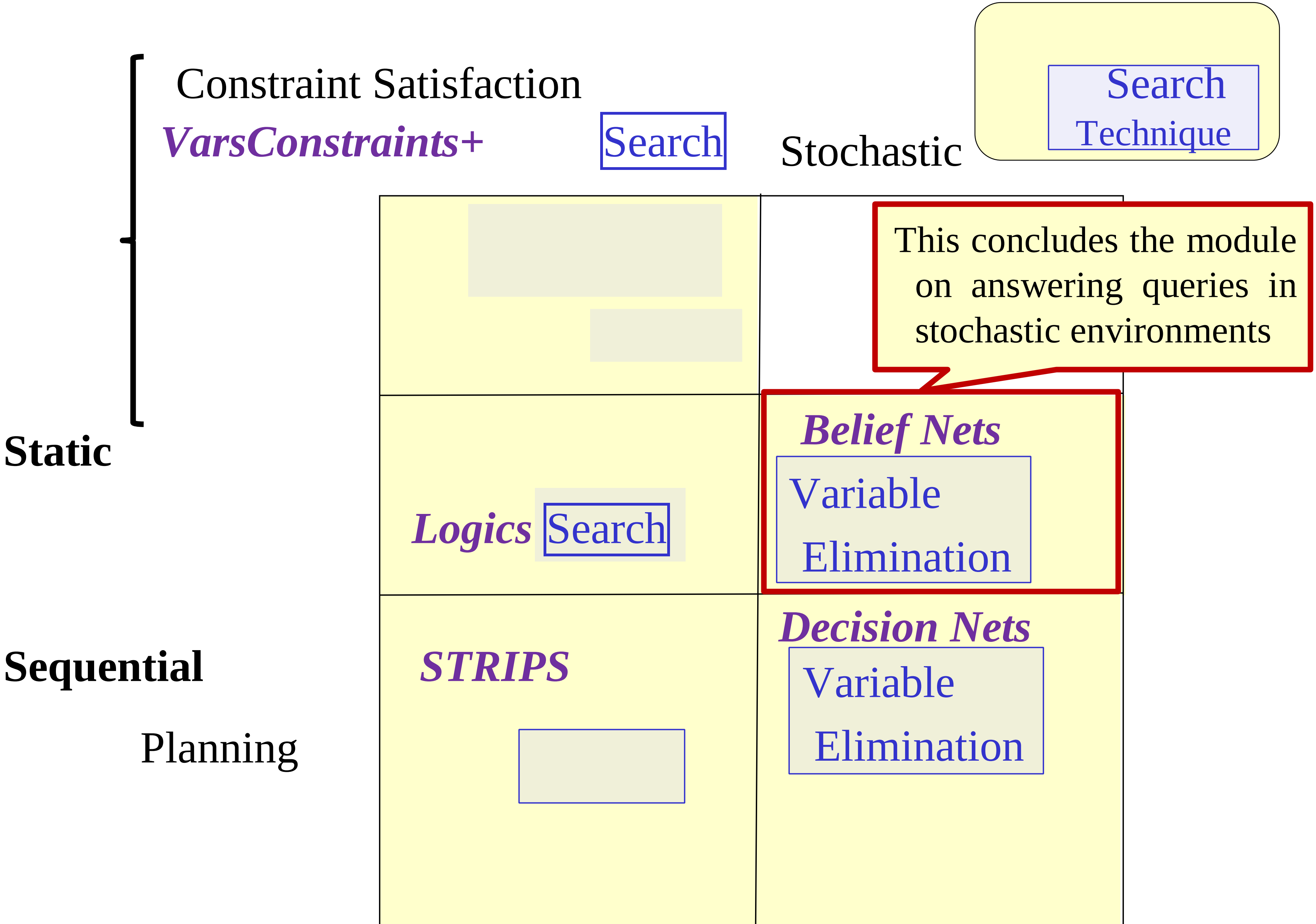
Deterministic

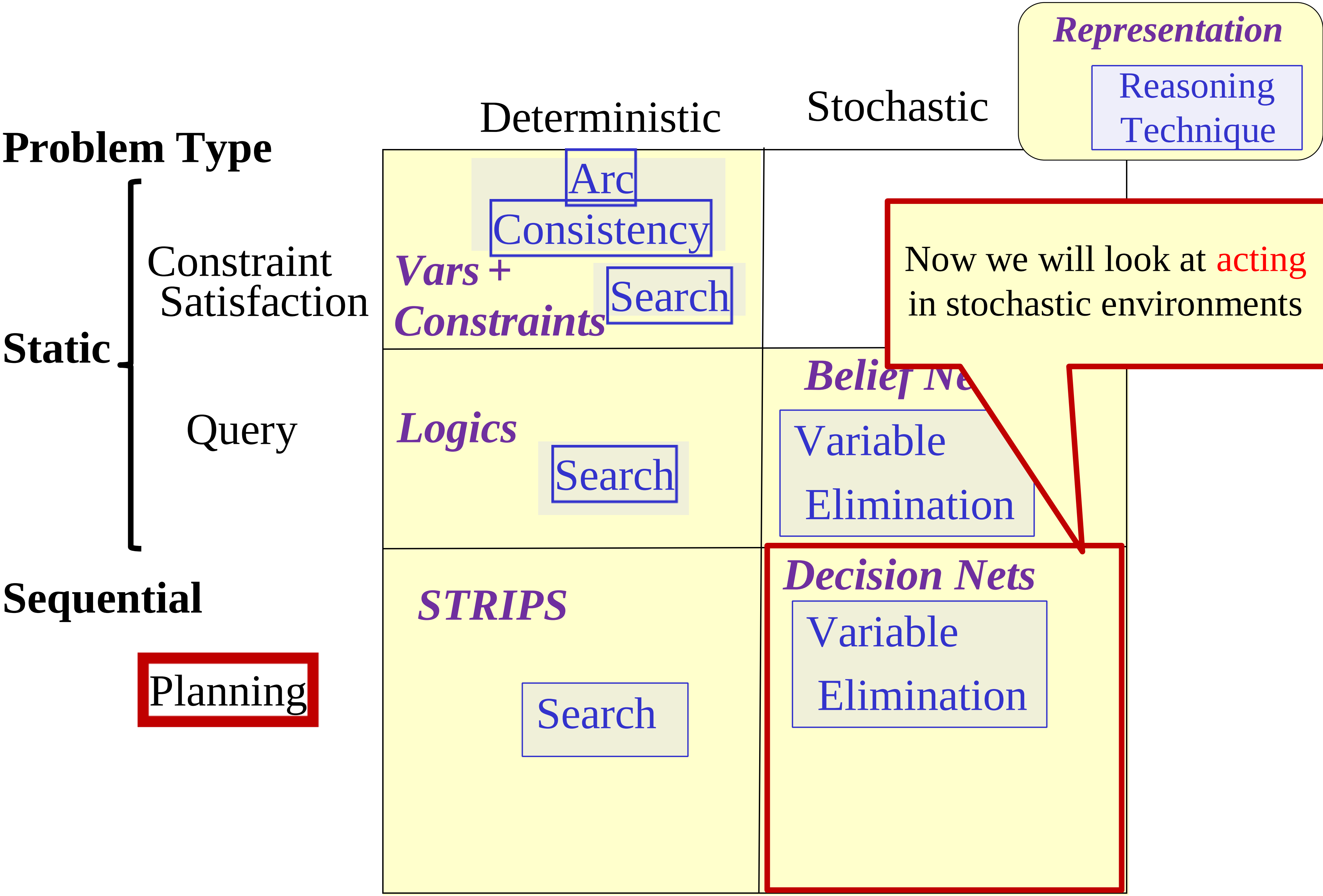
Problem Type

Arc

Consistency

Query


Constraint Satisfaction
VarsConstraints+
Search
Stochastic
Search
Technique
This concludes the module on answering queries in stochastic environments
Static
Belief Nets
Variable Elimination
Logics
Search
Sequential
STRIPS
Decision Nets
Variable Elimination
Planning



Representation
Reasoning Technique
Deterministic
Stochastic
Problem Type
Static
Constraint Satisfaction
Query
Sequential
Planning
Vars + Constraints
Arc Consistency
Search
Logics
Search
STRIPS
Search
Belief Ne
Variable Elimination
Decision Nets
Variable Elimination
Now we will look at acting in stochastic environments

Environment

Decisions Under Uncertainty: Intro

- Earlier in the course, we focused on decision making in deterministic domains
- Planning
- Now we face stochastic domains
- so far we've considered how to represent and update beliefs
- what if an agent has to make decisions (act) under uncertainty?

- Making decisions under uncertainty is important • We represent the world probabilistically so we can use our beliefs as the basis for making decisions

Decisions Under Uncertainty: Intro

- An agent's decision will depend on
- What actions are available
- What beliefs the agent has
- Which goals the agent has

- Differences between deterministic and stochastic setting • Obvious difference in representation: need to represent our uncertain beliefs

- Actions will be pretty straightforward: represented as decision variables
- Goals will be interesting: we'll move from all-or-nothing goals to a richer notion:
  - ✓rating how happy the agent is in different situations.
- Putting these together, we'll extend Bayesian Networks to make a new representation called Decision Networks

Delivery Robot Example

- Robot needs to reach a certain room
- Robot can go
- the short way - faster but with more obstacles, thus more prone to accidents that can damage the robot and prevent it from reaching the room

- the long way - slower but less prone to accident
- Which way to go? Is it more important for the robot to arrive fast, or to minimize the risk of damage?
- The Robot can choose to wear pads to protect itself in case of accident, or not to wear them. Pads make it heavier, increasing energy consumption
- Again, there is a tradeoff between reducing risk of damage, saving resources and arriving fast
- Possible outcomes
- No pad, no accident 
- Pad, no accident
- Pad, Accident
- No pad, accident

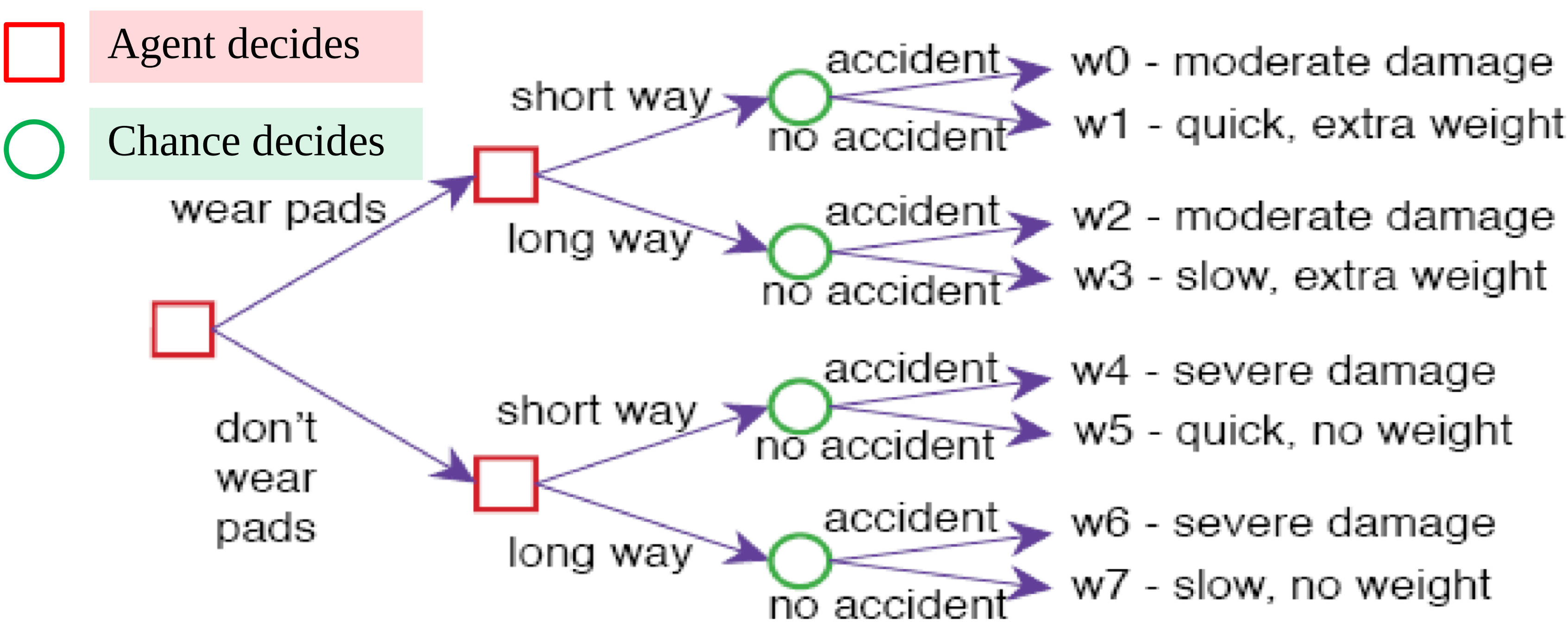
Next

- We'll see how to represent and reason about situations of this nature using Decision Trees, as well as
- Probability to measure the uncertainty in action outcome
- Utility to measure agent's preferences over the various outcomes
- Combined in a measure of expected utility that can be used to identify the action with the best expected outcome
- Best that an intelligent agent can do when it needs to act in a stochastic environment

Decision Tree for the Delivery Robot Example

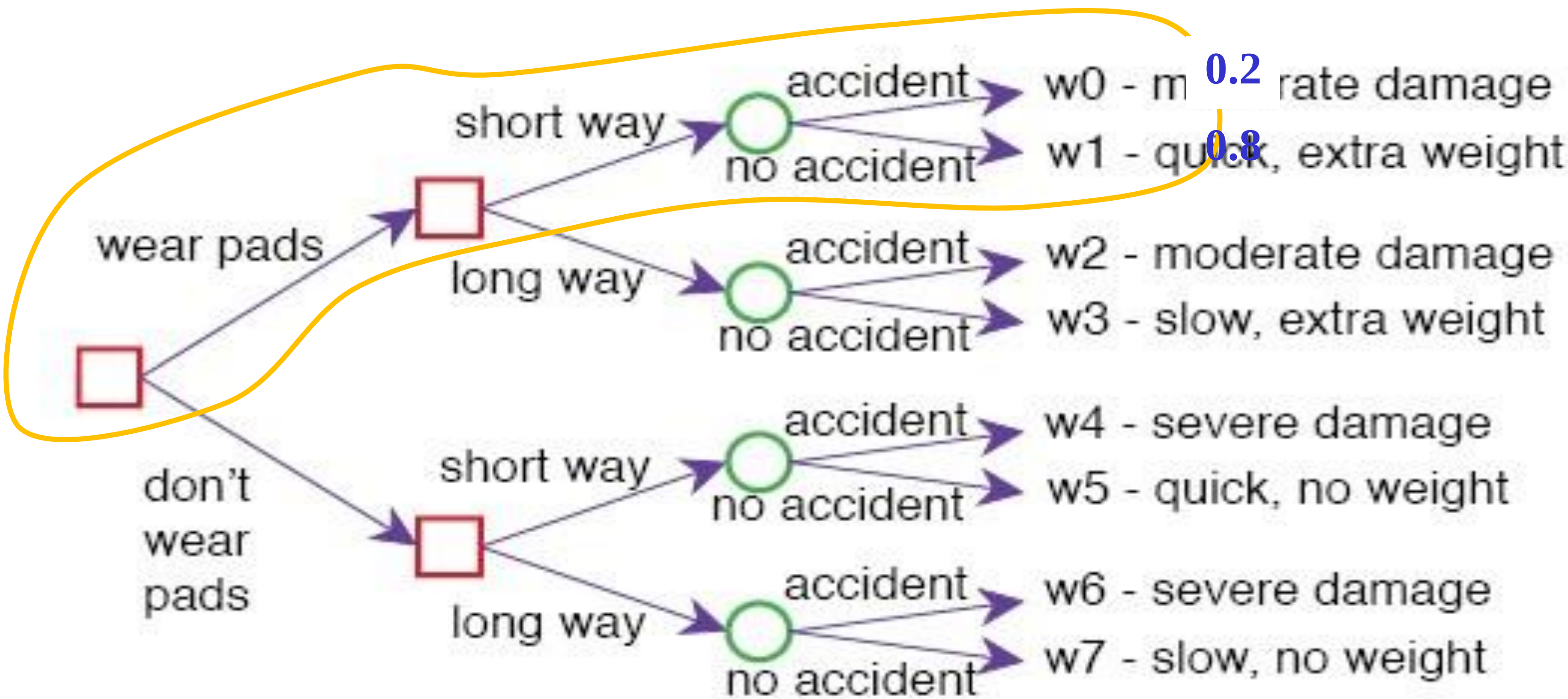
  - Decision variable 1: the robot can choose to wear pads

- Yes: protection against accidents, but extra weight
- No: fast, but no protection
- Decision variable 2: the robot can choose the way
- Short way: quick, but higher chance of accident
- Long way: safe, but slow
- Random variable: is there an accident?



Possible worlds and decision variables

- A possible world specifies a value for each random variable and each decision variable
  - For each assignment of values to all decision variables:

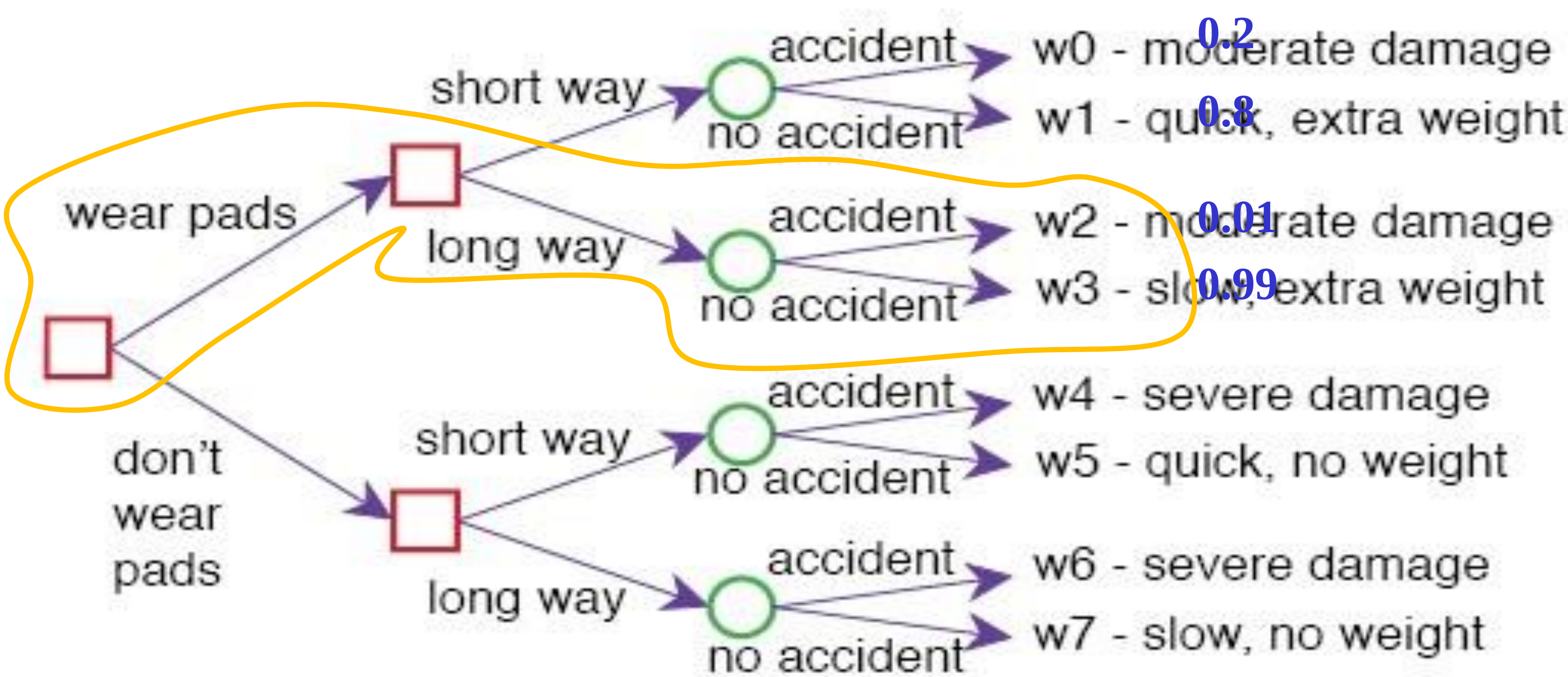


Possible worlds and decision variables

- A possible world specifies a value for each random variable
    - the probabilities of the worlds satisfying that assignment sum to 1.

Possible worlds and decision variables

- A possible world specifies a value for each random variable and each decision variable
  - For each assignment of values to all decision variables:

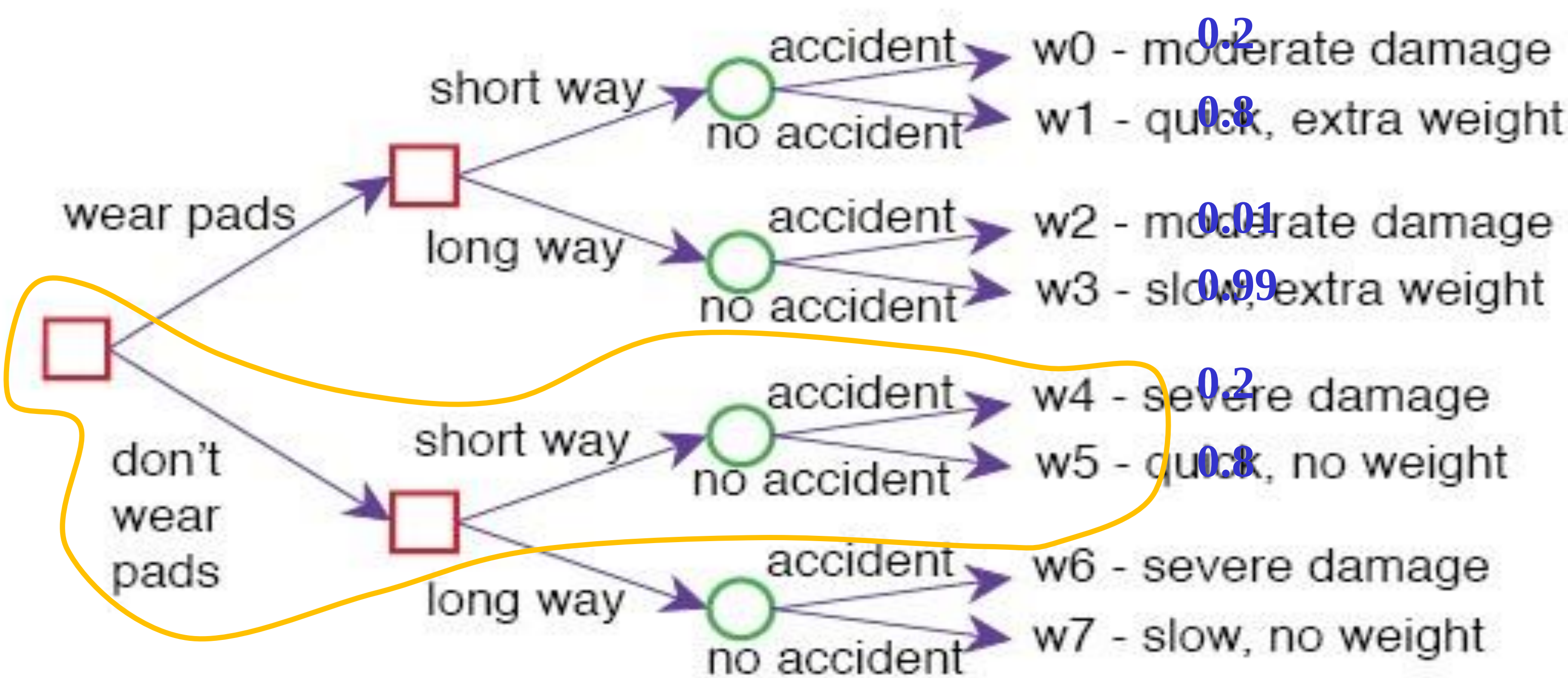


Possible worlds and decision variables

- A possible world specifies a value for each random variable
  - the probabilities of the worlds satisfying that assignment sum to 1.

Possible worlds and decision variables

- A possible world specifies a value for each random variable and each decision variable
  - For each assignment of values to all decision variables:

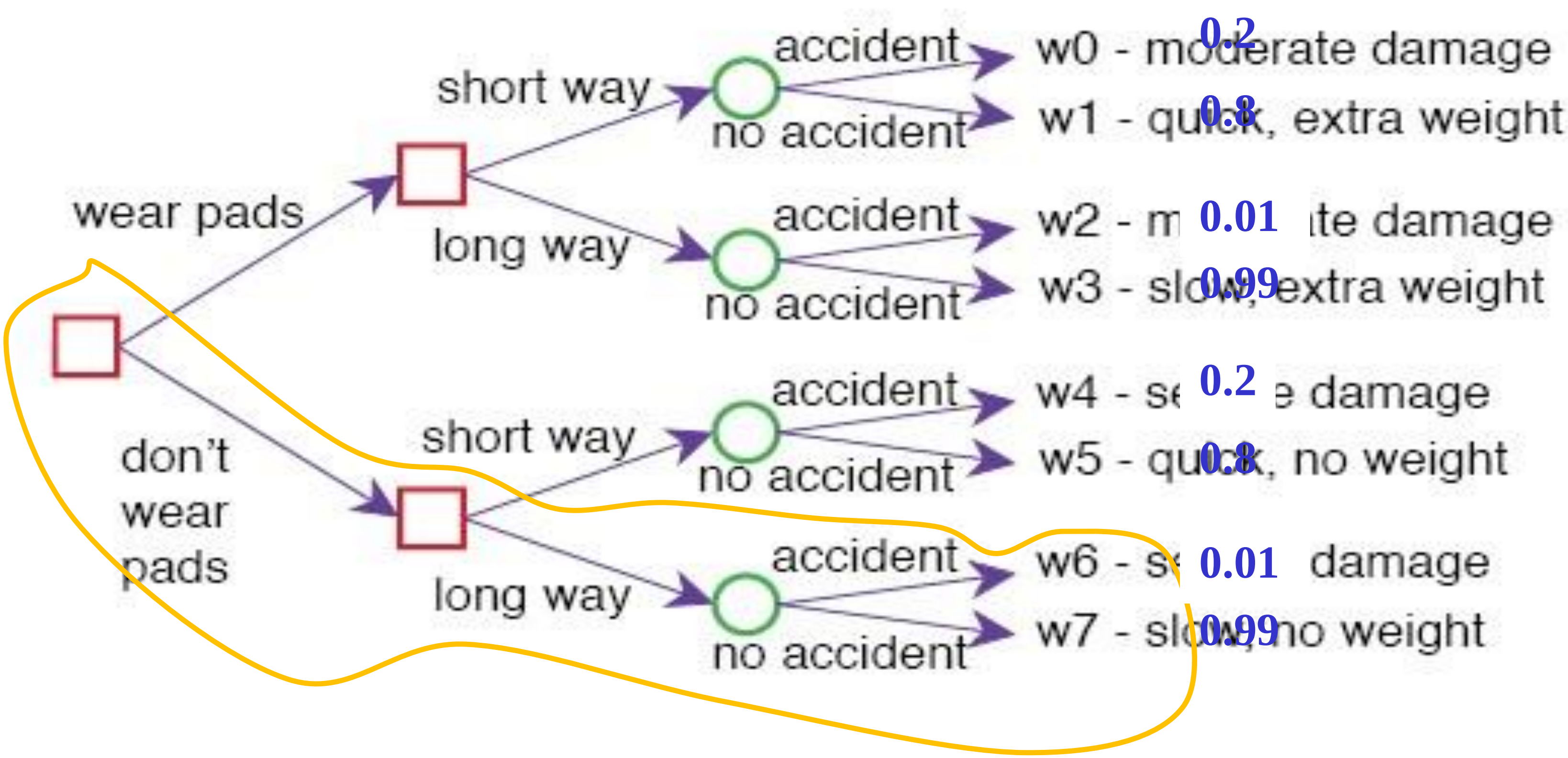


Possible worlds and decision variables

- A possible world specifies a value for each random variable
  - the probabilities of the worlds satisfying that assignment sum to 1.

Possible worlds and decision variables

- A possible world specifies a value for each random variable and each decision variable
  - For each assignment of values to all decision variables:



Possible worlds and decision variables

- A possible world specifies a value for each random variable
  - the probabilities of the worlds satisfying that assignment sum to 1.

Lecture Overview

- Recap
- Intro to Decision theory
- Utility and expected utility
- Decision Networks for single-stage decision problems

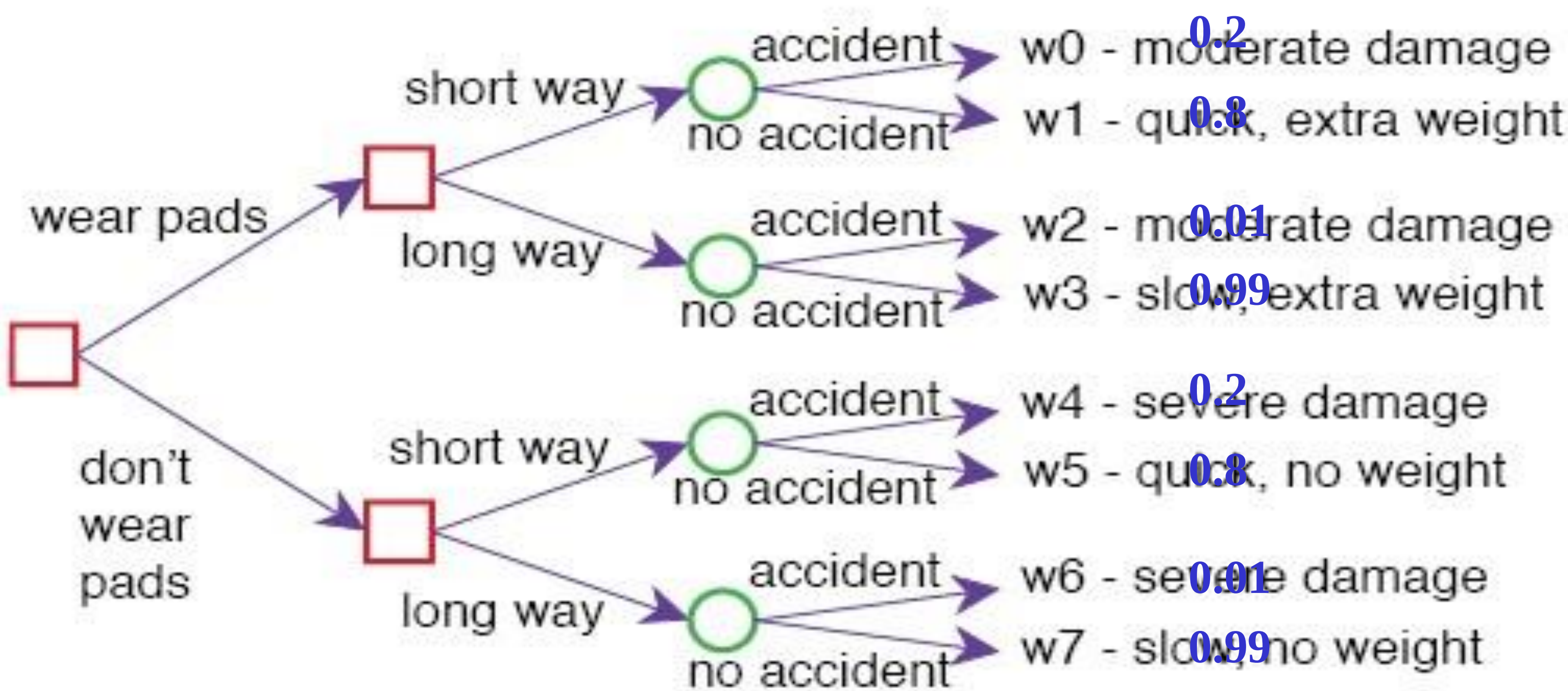
Utility

- 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
- Expressed by a number in [0,100]

Utility for the Robot Example

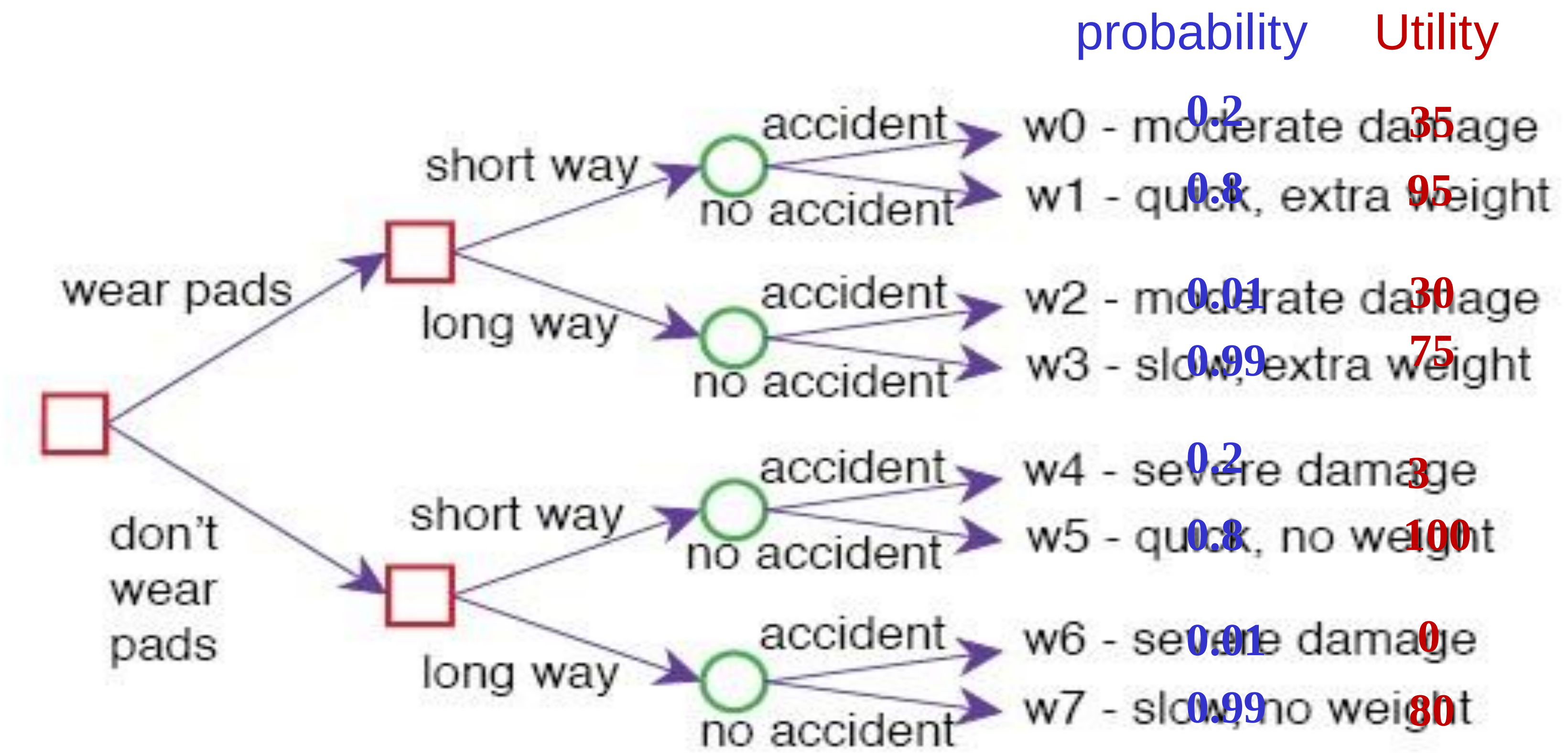
- Which would be a reasonable utility function for our robot?
- Which are the best and worst scenarios? probability

Utility



Utility for the Robot Example

- Which would be a reasonable utility function for our robot?


probability
Utility
wear pads
don't wear pads
short way
long way
short way
long way
accident
no accident
accident
no accident
accident
no accident
accident
no accident
w0 - moderate damage
w1 - quick, extra weight
w2 - moderate damage
w3 - slow, extra weight
w4 - severe damage
w5 - quick, no weight
w6 - severe damage
w7 - slow, no weight
0.2
35
0.8
95
0.01
30
0.99
75
0.2
3
0.8
100
0.01
0
0.99
80

Utility: Simple Goals

A.

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

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

Which way	Accident	Wear Pads	Utility

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

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 0

C .

D. Not possible

Utility

- 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
- Expressed by a number in [0,100]

- Simple goals can still be specified
- Worlds that satisfy the goal have utility 100 • Other worlds have utility 0

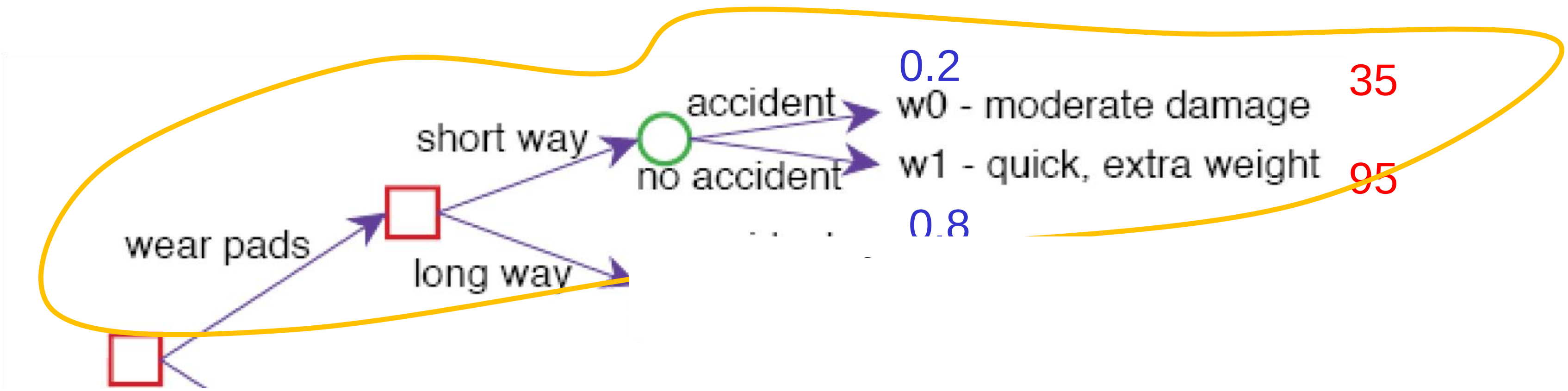
e.g., goal "reach the room"

Optimal decisions: combining Utility and Probability

Which way	Accident Pads	Wear	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

- Each set of decisions defines a probability distribution over possible outcomes
- Each outcome has a utility
- For each set of decisions, we need to know their expected utility

- the value for the agent of achieving a certain probability distribution over outcomes (possible worlds)
- The expected utility of a set of decisions is obtained by
- weighting the utility of the relevant possible worlds by their probability.



- We want to find the decision with maximum expected utility

Expected utility of a decision

- The expected utility of a specific decision $D = d$ is indicated as $E(U \mid D = d)$ and it is computed as follows

$$P(w_1)\times U(w_1) + P(w_2)\times U(w_2) + P(w_3)\times U(w_3) + \ldots + P(w_n)\times U(w_n)$$

Where

- $w_1, w_2, w_3 \ldots, w_n$ are all the possible worlds in which d is true
- $P(w_1), P(w_2), \ldots P(w_n)$ are their probabilities
- $U(w_1), U(w_2), \ldots U(w_n)$ are the corresponding utilities for $w_1, \ldots, w_n$

That is,

- for each possible world w_i in which the decision d is true, multiply the probability of that world and its utility $P(w_i) \times U(w_i)$

- sum all these products together

This notation indicates all the possible worlds in which d_i is true

In a formula

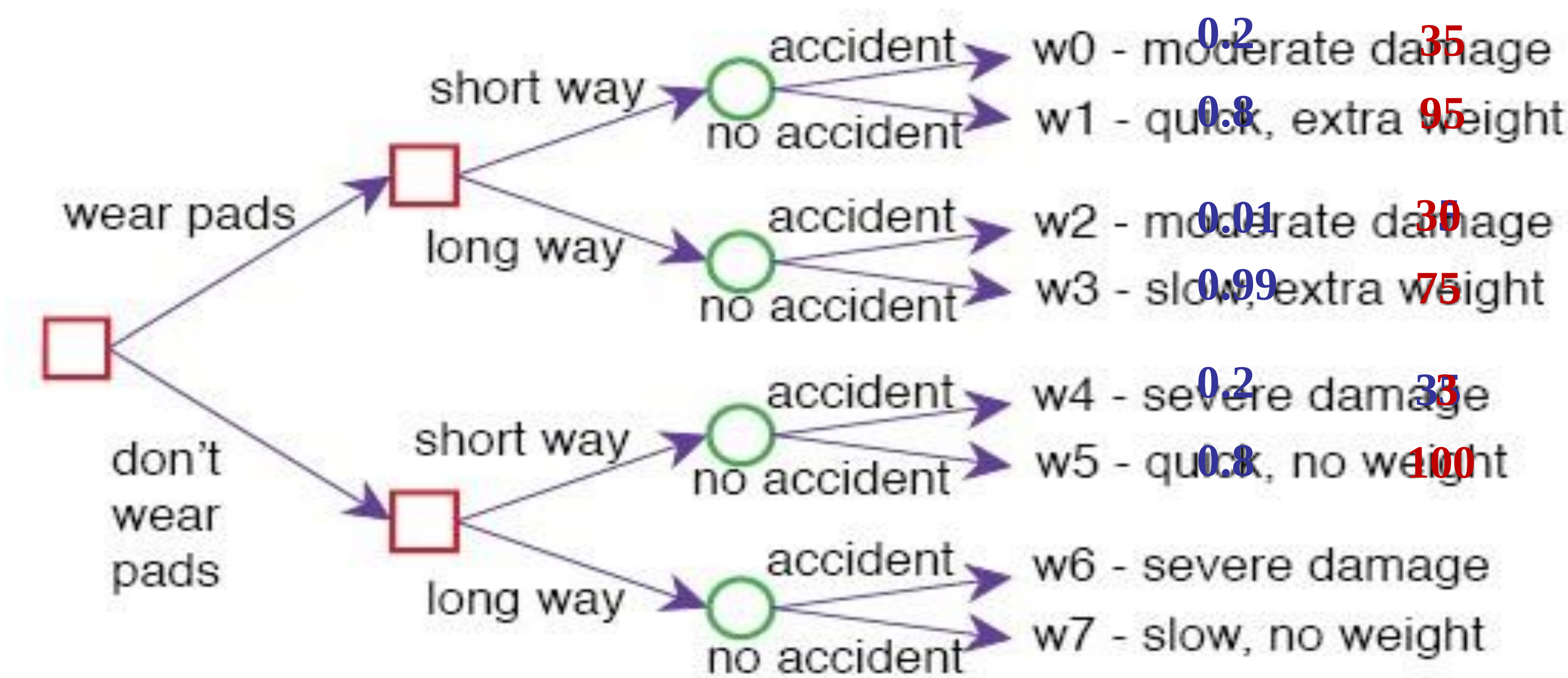
$$E(U \mid D = d_i) = \sum_{w \models (D = d_i)} P(w) \times U(w)$$

Example of Expected Utility

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

$$E(U \mid D = d) = \sum_{w \models (D = d)} P(w)\, U(w) = \mathbf{P(w_1) \times U(w_1) + \ldots. + P(w_n) \times U(w_n)}$$

- What is the expected utility of Wearpads=yes, Way=short ?



A. 7 B. 83 C. 76 D. 157.55

Probability Utility E[U|D]

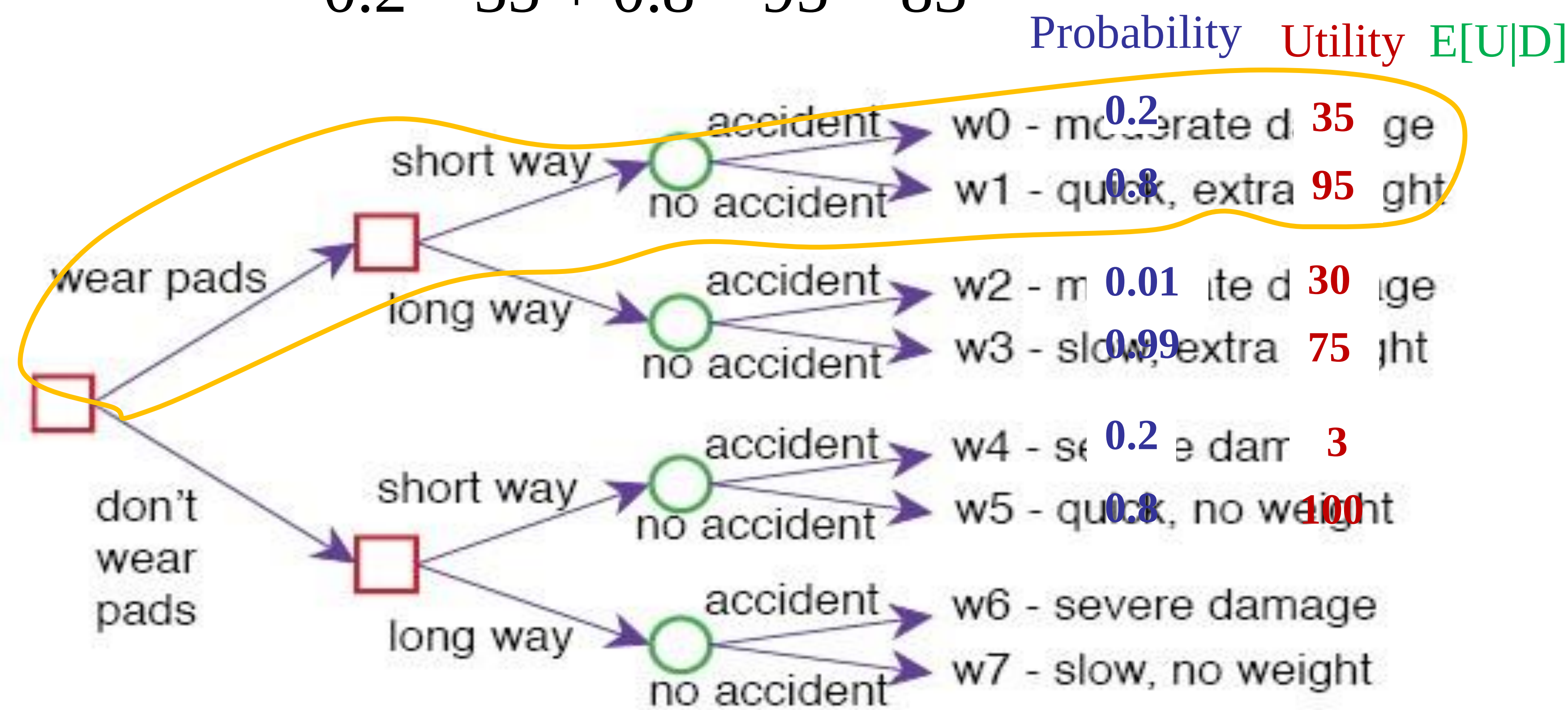
0.01350 0.99 80

Expected utility of a decision

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

$$E(U \mid D = d) = \sum_{w \models (D = d)} P(w)\, U(w) = \mathbf{P(w_1) \times U(w_1) + \ldots . + P(w_n) \times U(w_n)}$$

- $0.2 * 35 + 0.8 * 95 = 83$



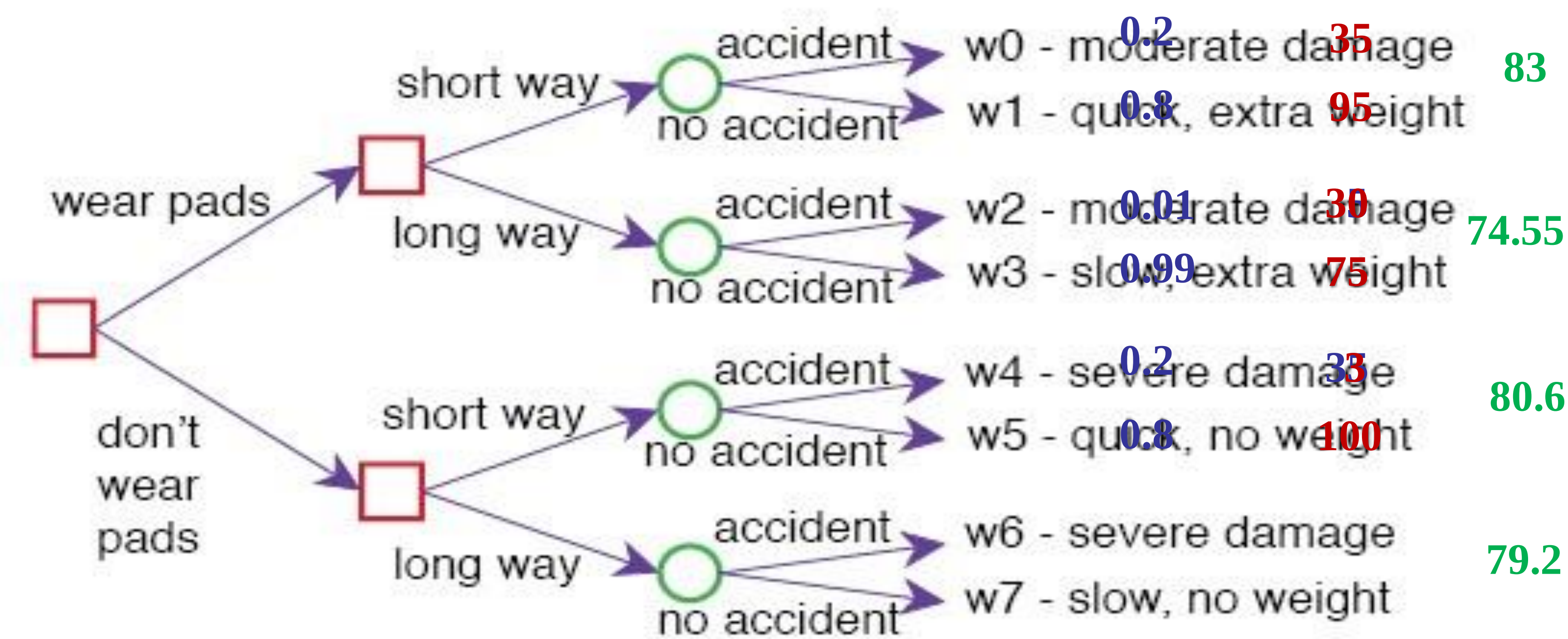
- What is the expected utility of Wearpads=yes, Way=short?

0.01350 0.99 80

Expected utility of a decision

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

$$E(U \mid D = d) = \sum_{w \models (D = d)} P(w)\, U(w) = \mathbf{P(w_1) \times U(w_1) + \ldots + P(w_n) \times U(w_n)}$$



Probability Utility E[U|D]

0.01350 0.99 80

Lecture Overview

- Recap
- Intro to Decision theory
- Utility and expected utility
- Decision Networks for Single-stage decision problems

Single Action vs. Sequence of Actions

- Single Action (aka One-Off Decisions)
  - One or more primitive decisions that can be treated as a single macro decision to be made before acting
  - E.g., “WearPads” and “WhichWay” can be combined into macro decision (WearPads, WhichWay) with domain {yes, no} × {long, short}

- Sequence of Actions (Sequential Decisions)
  - Repeat:
    - ✓make observations
    - ✓decide on an action
    - ✓carry out the action

- Agent has to take actions not knowing for sure what the future brings
  - ✓This is fundamentally different from everything we've seen so far
  - ✓Planning was sequential, but agent could still think first and then act

Optimal single-stage decision

Given a single (macro) decision variable D

- the agent can choose $D=d_i$ for any value $d_i \in dom(D)$

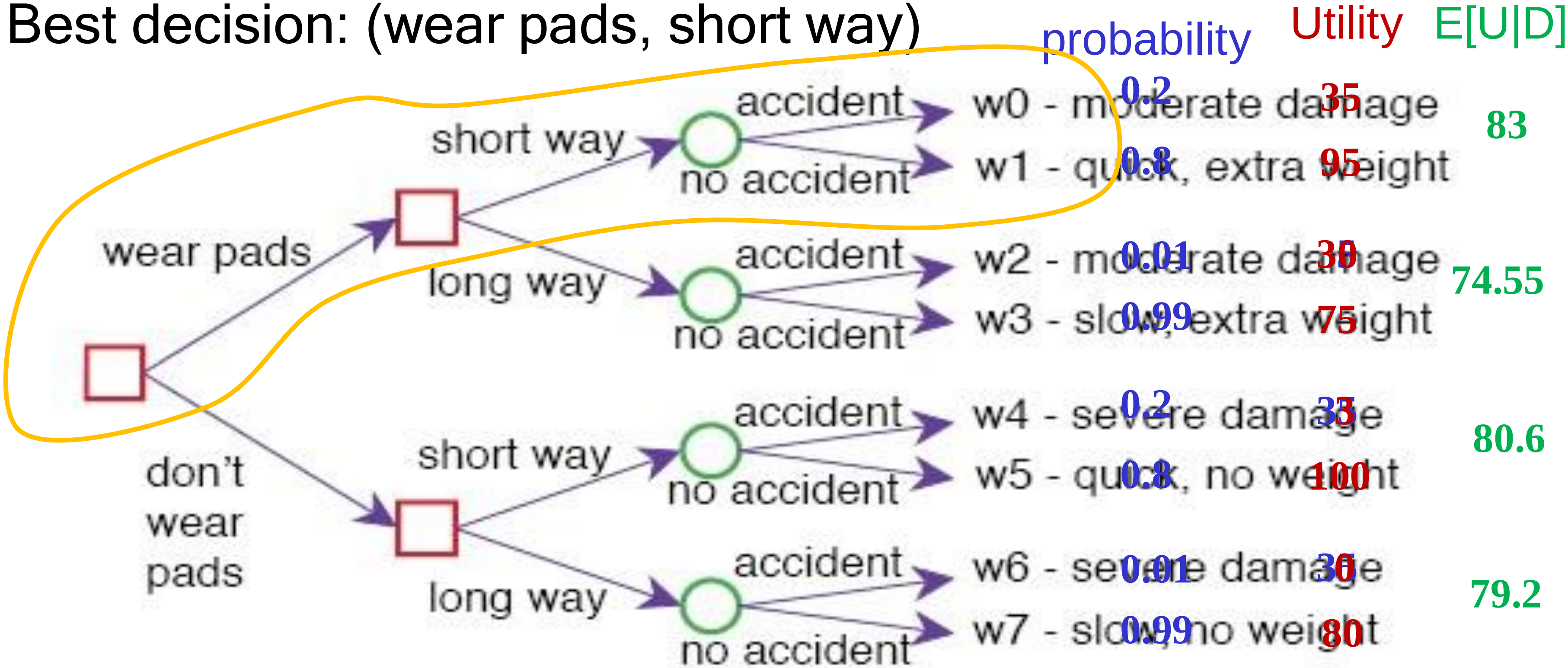
Definition (optimal single-stage decision)

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

$$d_{max} \in \underset{d_i \in dom(D)}{\operatorname{argmax}} E[U|D=d_i]$$

Optimal decision in robot delivery example

Best decision: (wear pads, short way)



Definition (optimal single-stage decision)

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

$$d_{max} \in \underset{d_i \in dom(D)}{\operatorname{argmax}} \; E[U | D = d_i]$$

Conditional

Single-Stage decision networks

Extend belief networks

Random variables: same as in Bayesian networks

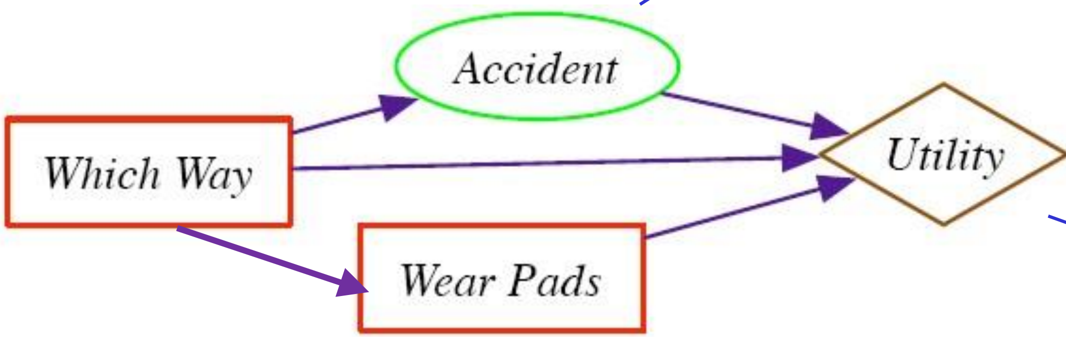
- drawn as an ellipse
- Arcs into the node represent probabilistic dependence
- random variable is conditionally independent of its non-descendants gi its parents

Decision nodes, that the agent chooses the value for

- Parents: only other decision nodes allowed
  - ✓represent information available when the decision is made

- Domain is the set of possible actions
- Drawn as a rectangle Exactly one utility node
- Parents: all random & decision variables on which the utility depends
- Specifies a utility for each instantiation of its parents
- Drawn as a diamond

Explicitly shows dependencies.
E.g., which variables affect the probability of an accident and the agent's utility?



Example Decision Network

Which Way W	Accident A	P(A|W)
long	true false	0.01
long	true false	0.99
short		0.2
short		

Which way Pads	Accident	Wear	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

0.8

Which Way

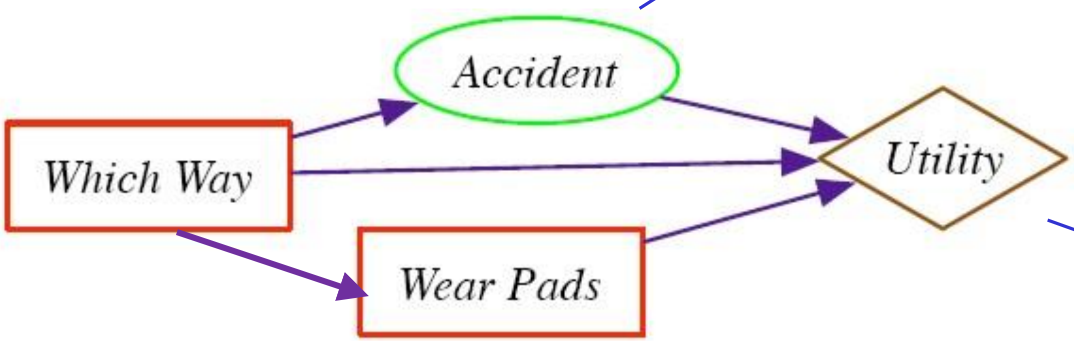
Example Decision Network

t
f

Decision nodes simply list the available decisions.

Which Way W	Accident A	P(A|W)

Explicitly shows dependencies. E.g., which variables affect the probability of an accident and the agent's utility?



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

long	true false	0.01
long	true false	0.99
short		0.2
short		0.8

Decision nodes simply list the available decisions.

Applet for Bayesian and Decision Networks

Which Way
t
f

The Belief and Decision Networks we have seen previously allows you to load predefined Decision networks for various domains and run queries on them.

Select one of the available examples via “File -> Load Sample Problem For Decision Networks

- Choose any of the examples below the blue line in the list that appears
- Right click on a node to perform any of these operations
- View the CPT/Decision table/Utility table for a chance/decision/utility node
- Make an observation for a chance variable (i.e., set it to one of its values)

- Query the current probability distribution for a chance node given the observations made
- A dialogue box will appear the first time you do this. Select “Always brief” at the bottom, and then click “Brief”.
- To compute the optimal decision (policy) click on the “Optimize Decision” button in the toolbar and select Brief in the dialogue box that will appear
- To see the actual policy, view the decision table for each decision node in the network

See available help pages and video tutorials for more details on how to use the Bayes applet (http://www.aispace.org/bayes/)

Learning Goals for Decision Under Uncertainty so Far

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