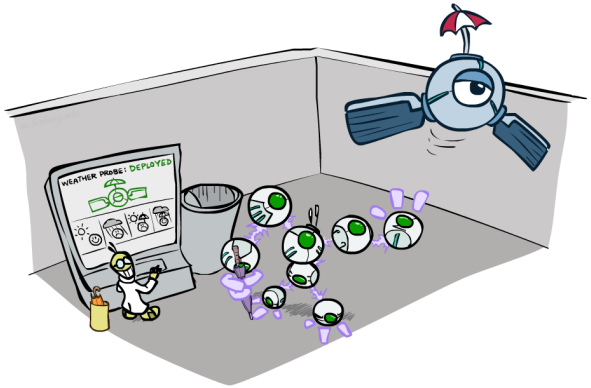
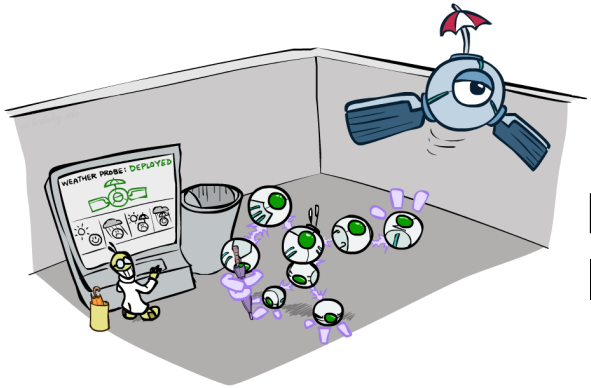
CS 188: Artificial Intelligence

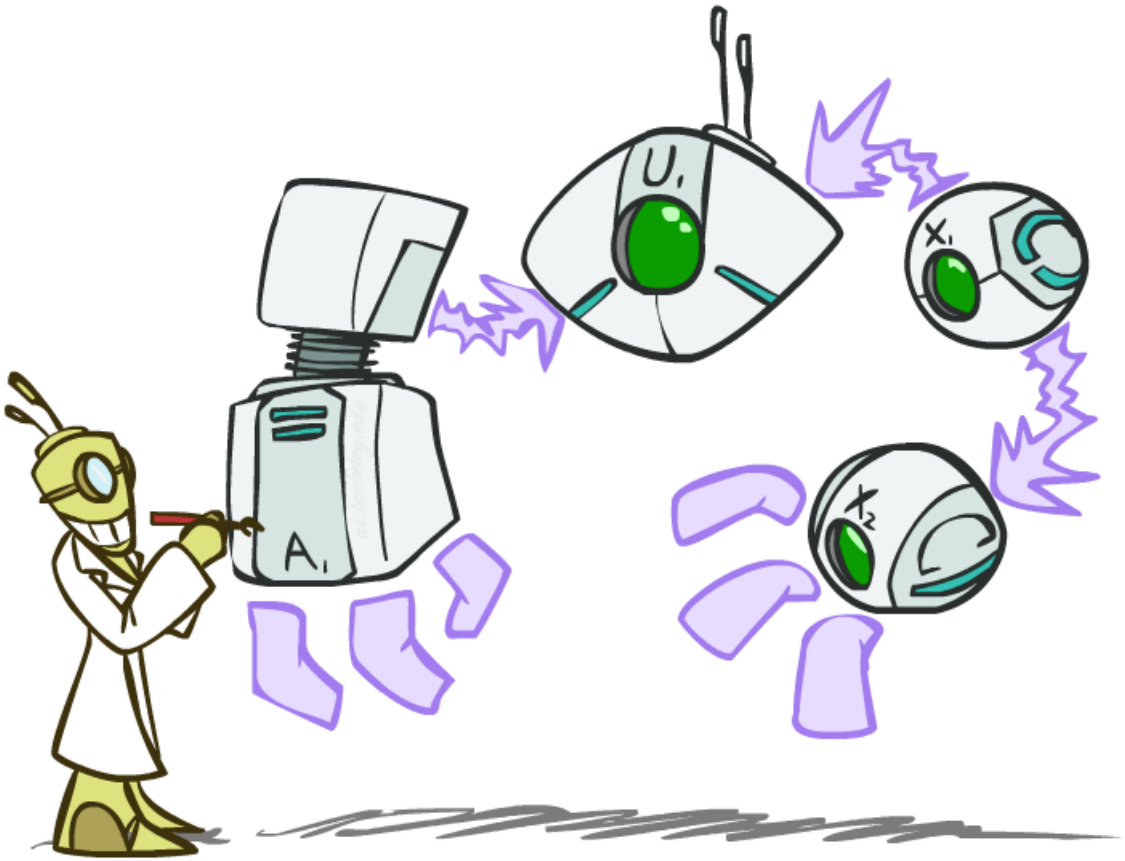


CS 188: Artificial Intelligence

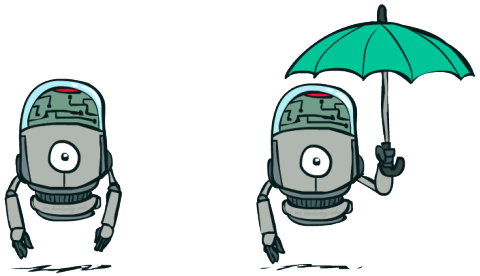


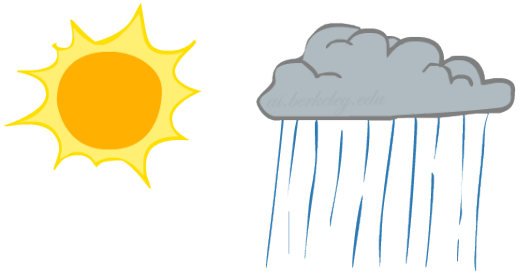
Decision Networks and Value of Perfect Information

Decision Networks

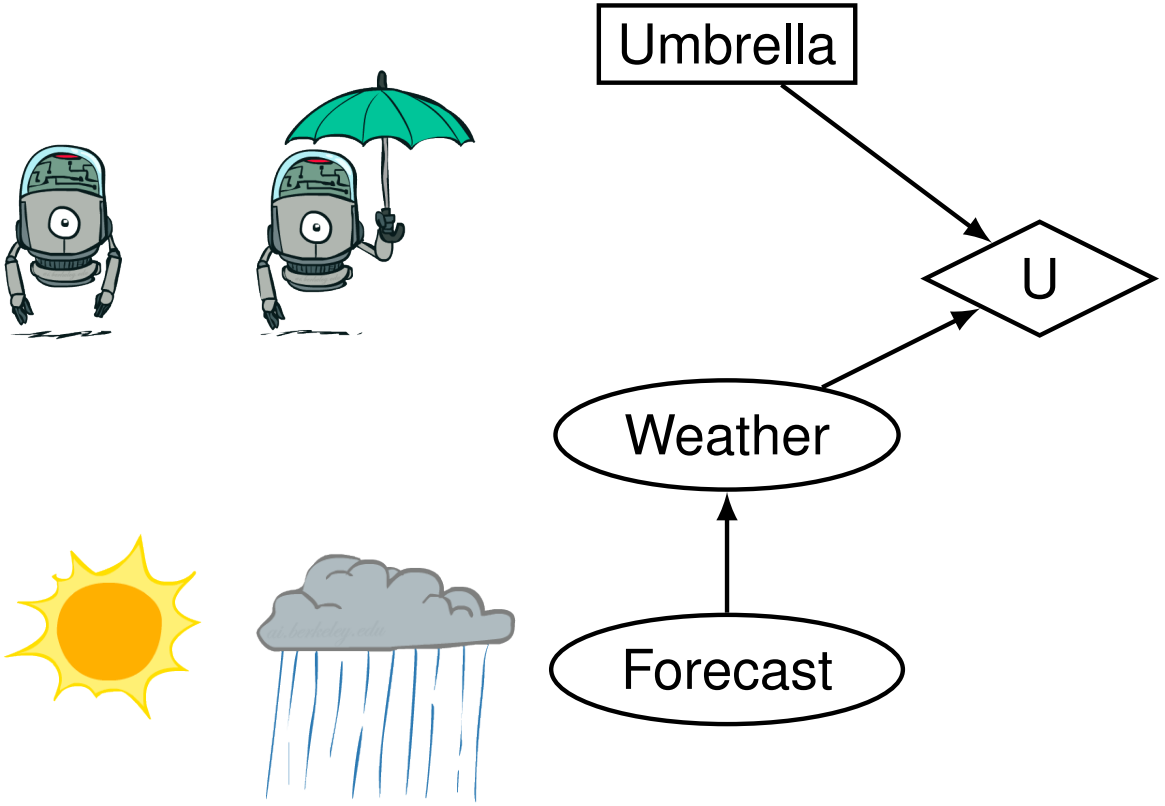


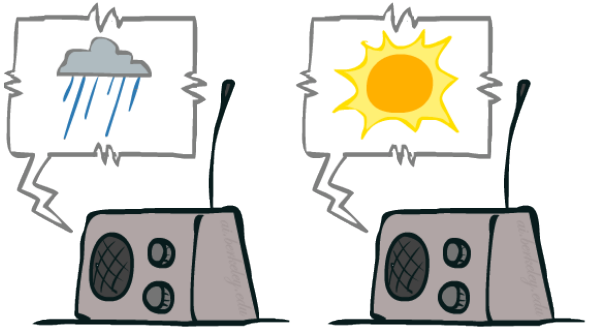
Decision Networks

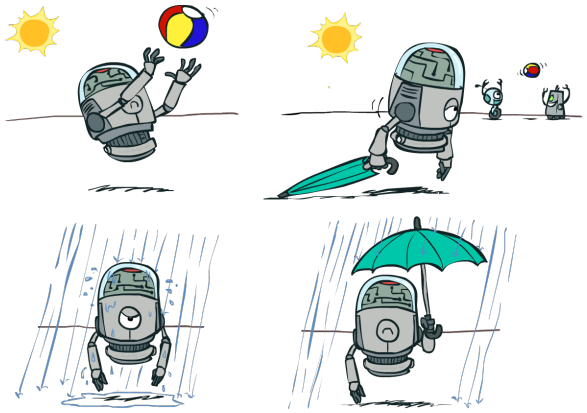




Decision Networks







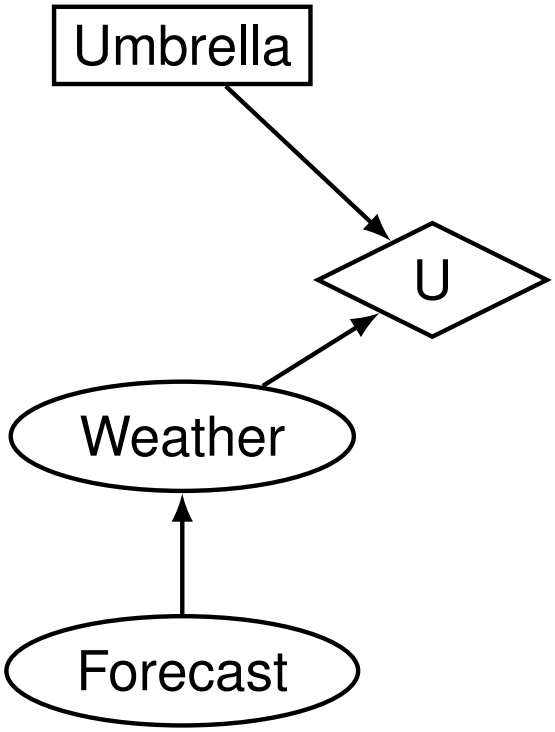
Decision Networks

Decision Networks

MEU: choose action which maximizes expected utility given evidence

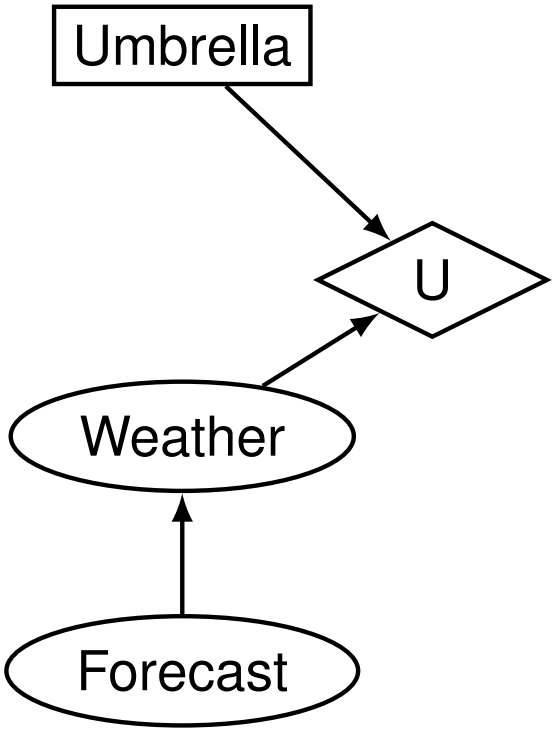
Decision Networks

MEU: choose action which maximizes expected utility given evidence



Decision Networks

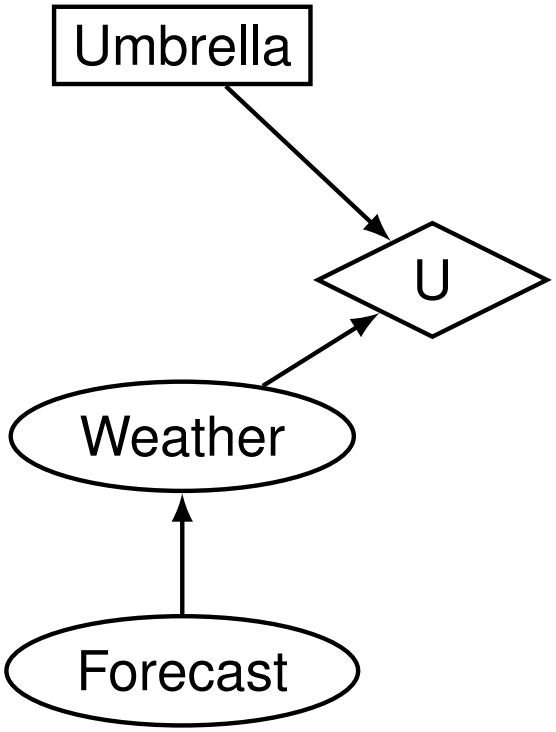
MEU: choose action which maximizes expected utility given evidence



Directly operationalize with decision networks

Decision Networks

MEU: choose action which maximizes expected utility given evidence

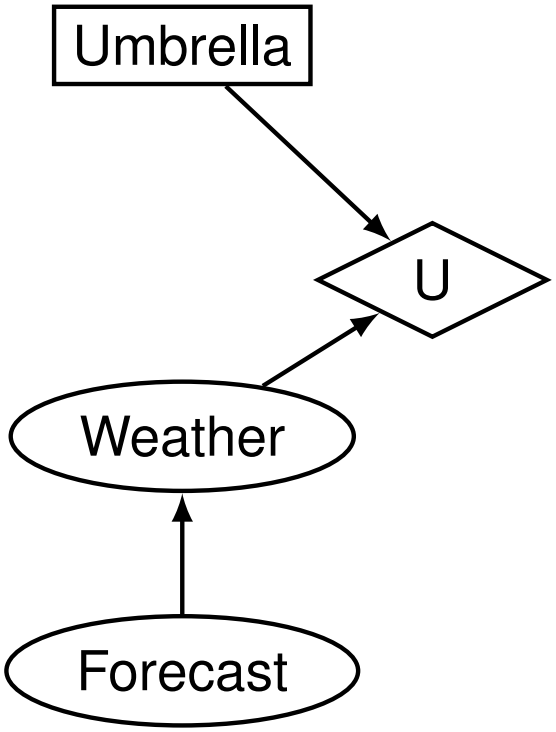


Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions

Decision Networks

MEU: choose action which maximizes expected utility given evidence

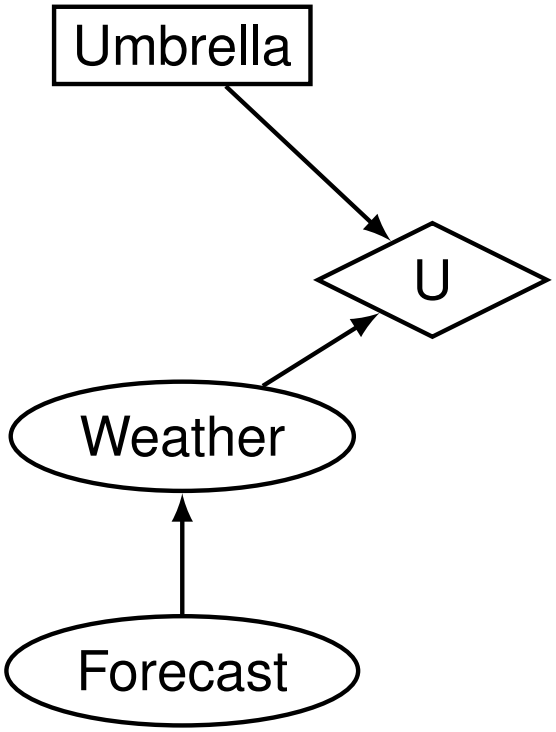


Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

Decision Networks

MEU: choose action which maximizes expected utility given evidence

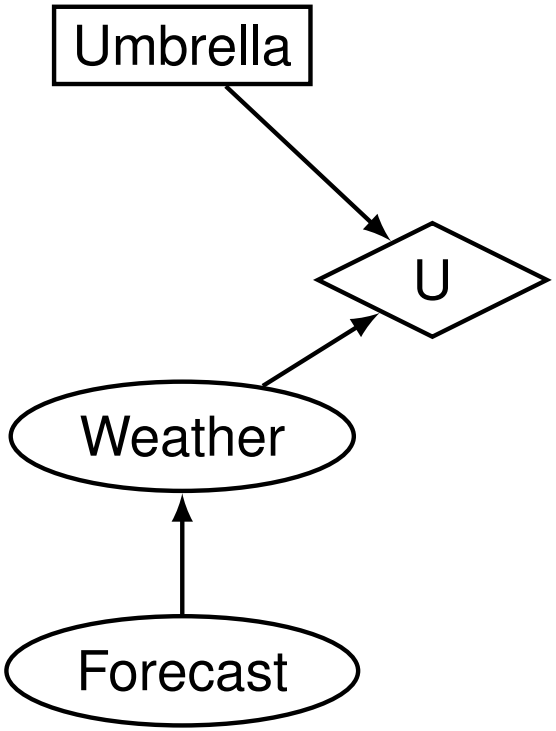


Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

Decision Networks

MEU: choose action which maximizes expected utility given evidence



Directly operationalize with decision networks

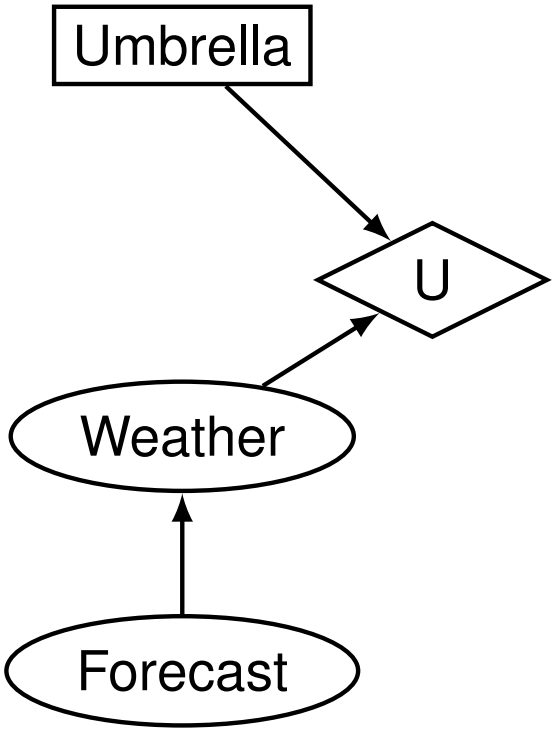
- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

New node types:

Chance nodes (just like BNs)

Decision Networks

MEU: choose action which maximizes expected utility given evidence



Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

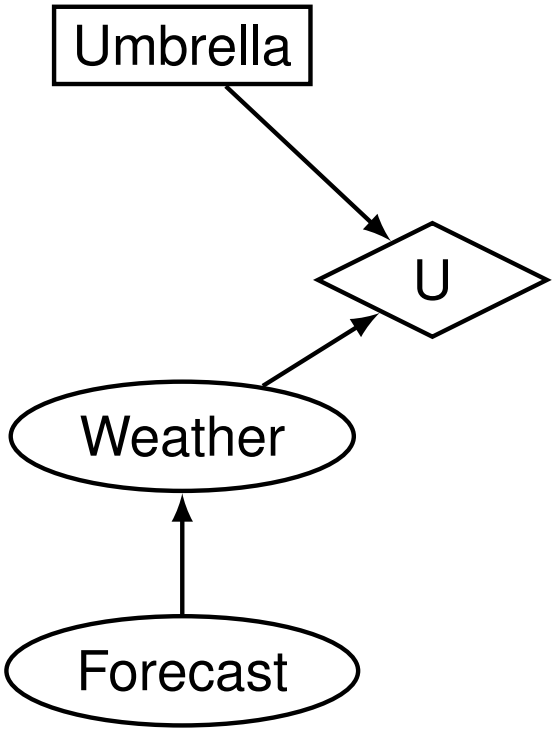
New node types:

Chance nodes (just like BNs)

Actions (rectangles, no parents, acts as observed evidence)

Decision Networks

MEU: choose action which maximizes expected utility given evidence



Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

New node types:

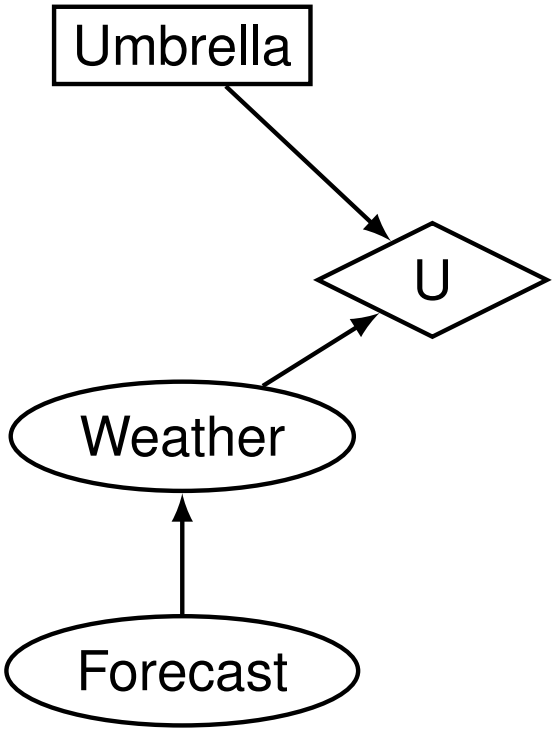
Chance nodes (just like BNs)

Actions (rectangles, no parents, acts as observed evidence)

Utility node (diamond, depends on action/chance nodes)

Decision Networks

MEU: choose action which maximizes expected utility given evidence



Directly operationalize with decision networks

- Bayes nets with nodes for utility and actions
- Can calculate expected utility for actions

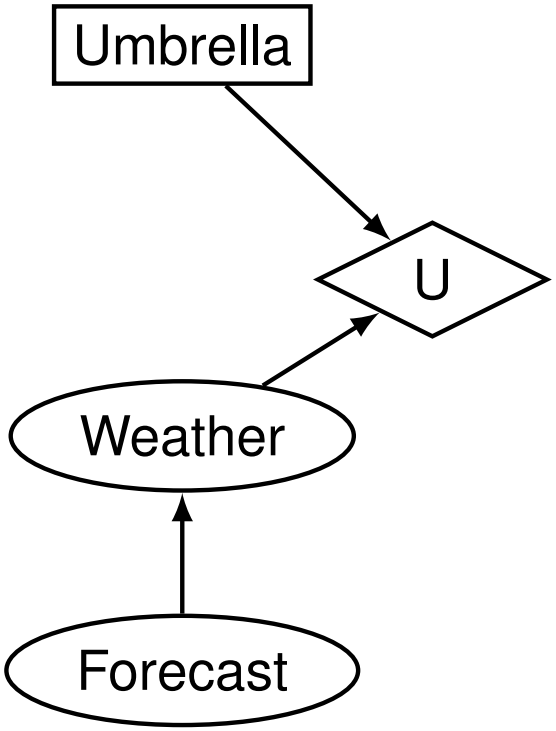
New node types:

Chance nodes (just like BNs)

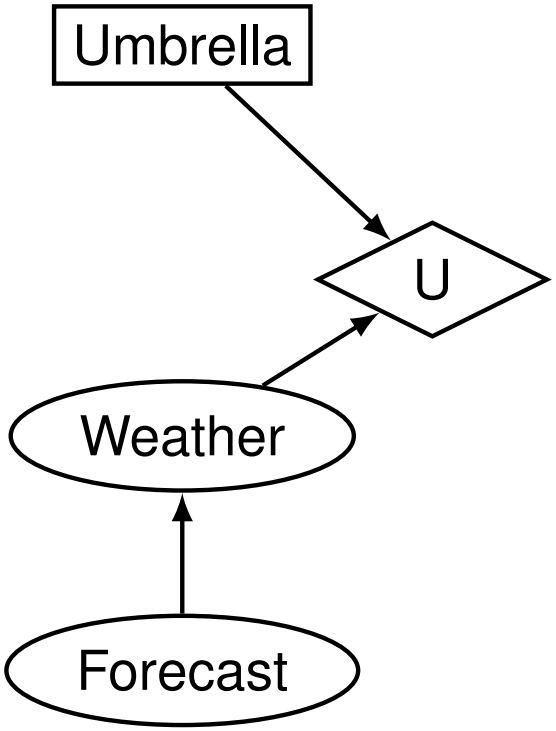
Actions (rectangles, no parents, acts as observed evidence)

Utility node (diamond, depends on action/chance nodes)

Decision Networks

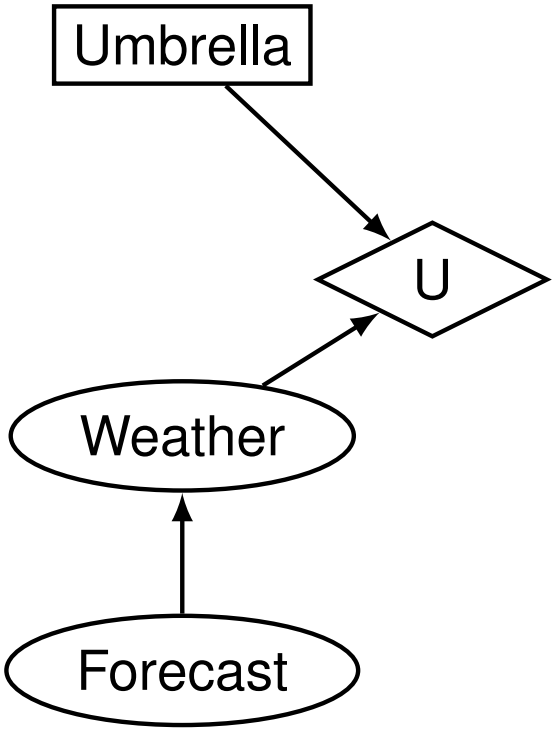


Decision Networks



Action selection

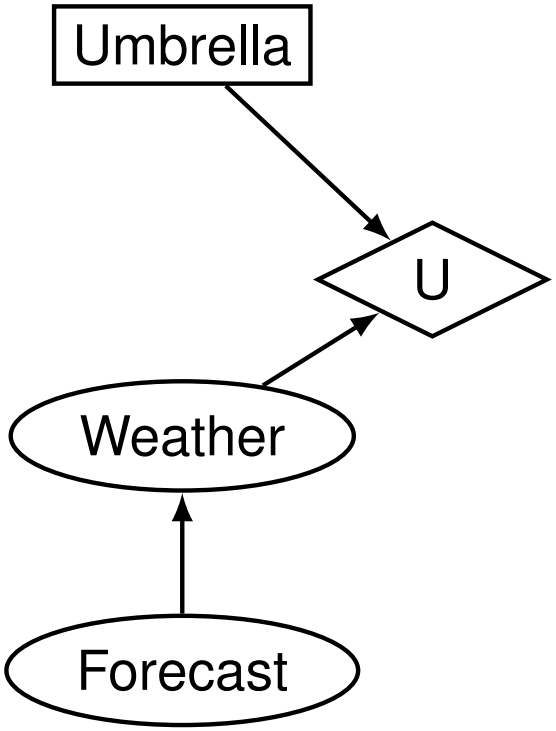
Decision Networks



Action selection

- Instantiate all evidence

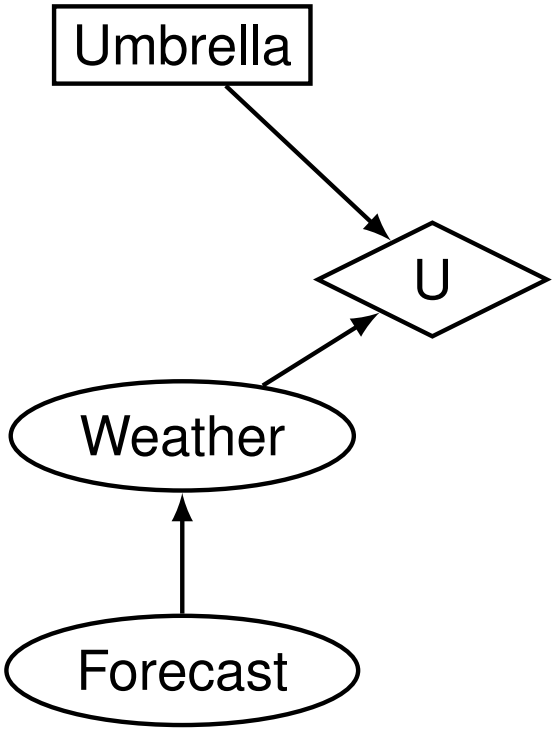
Decision Networks



Action selection

- Instantiate all evidence
- Set action node(s) each possible way

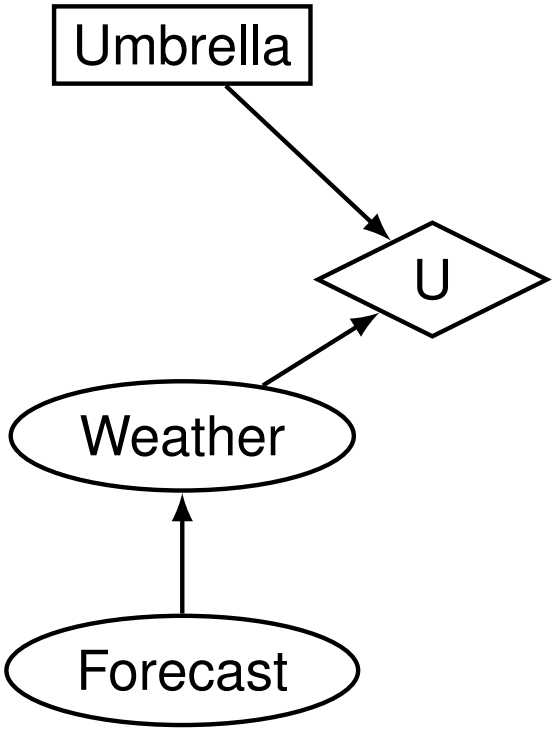
Decision Networks



Action selection

- Instantiate all evidence
- Set action node(s) each possible way
- Calculate posterior for parents of utility node, given evidence

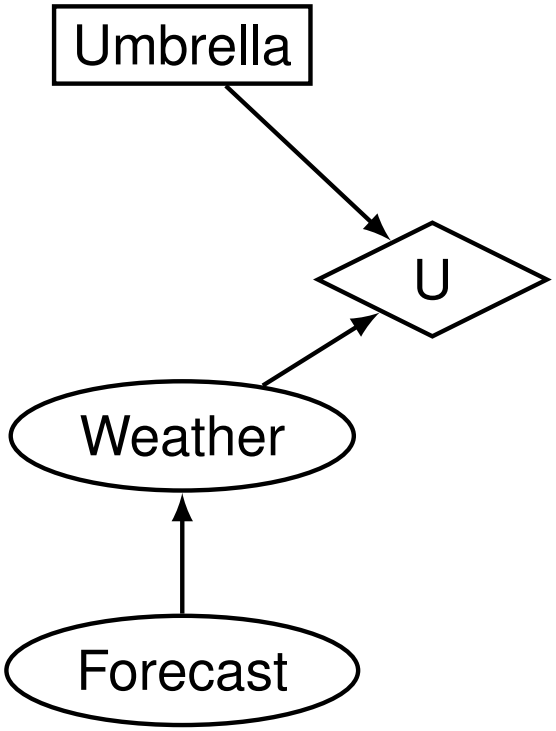
Decision Networks



Action selection

- Instantiate all evidence
- Set action node(s) each possible way
- Calculate posterior for parents of utility node, given evidence
- Calculate expected utility for each action

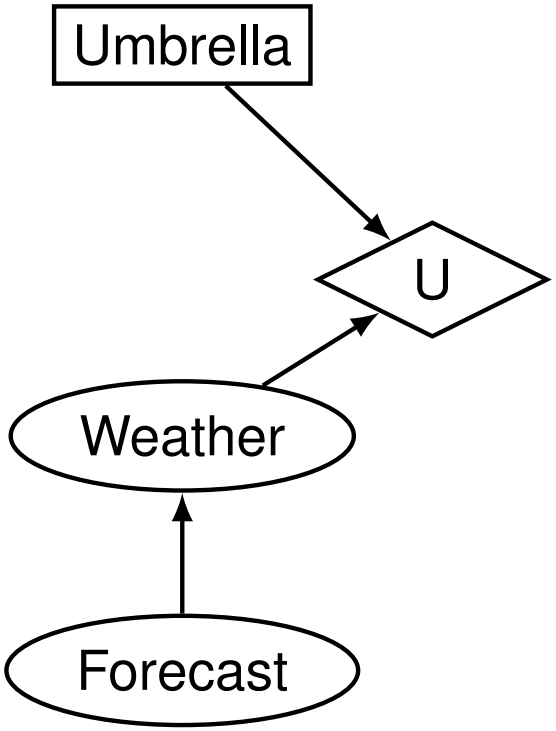
Decision Networks



Action selection

- Instantiate all evidence
- Set action node(s) each possible way
- Calculate posterior for parents of utility node, given evidence
- Calculate expected utility for each action
- Choose maximizing action

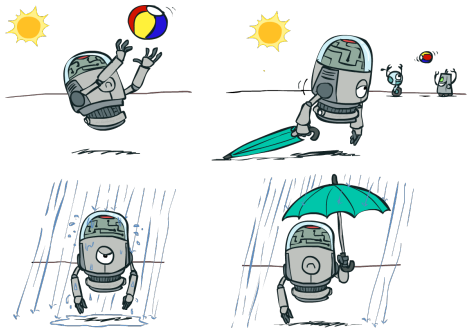
Decision Networks

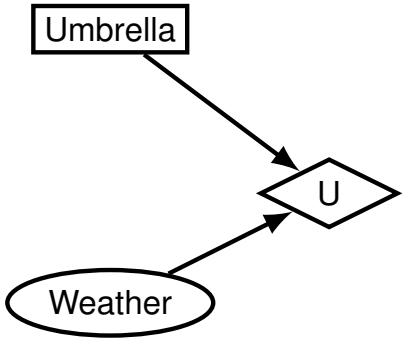


Action selection

- Instantiate all evidence
- Set action node(s) each possible way
- Calculate posterior for parents of utility node, given evidence
- Calculate expected utility for each action
- Choose maximizing action

Decision Networks

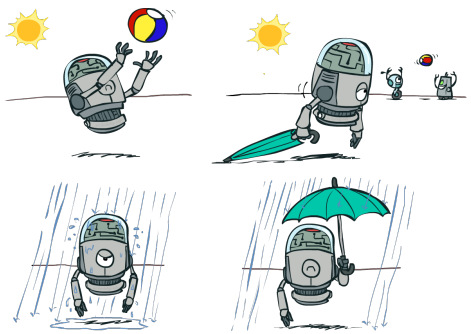


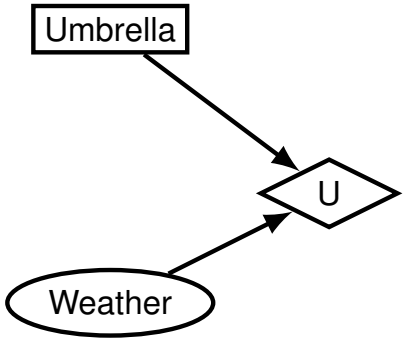


W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Decision Networks



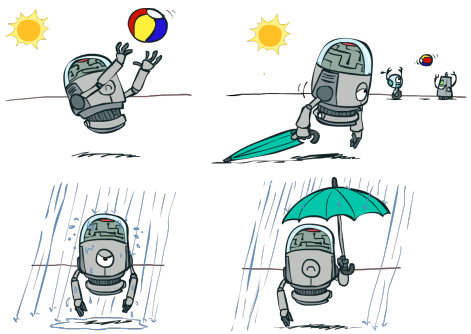


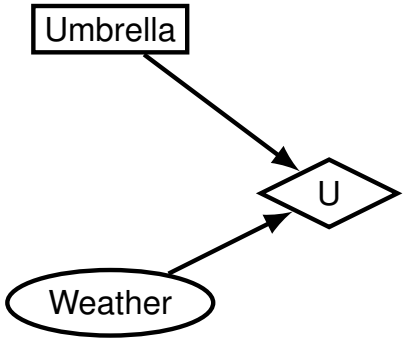
W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

Decision Networks





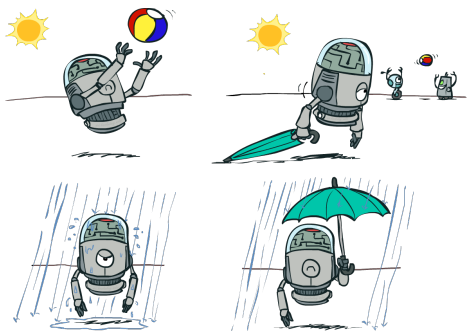
W	P(W)
sun	0.7
rain	0.3

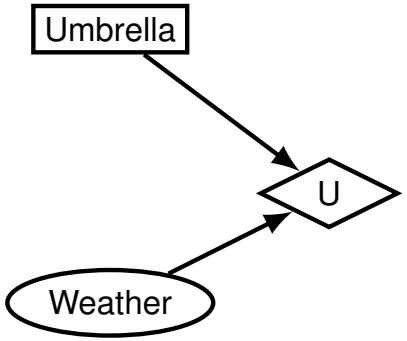
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

$$EU(leave) = \sum_w P(w)U(leave, w)$$

Decision Networks





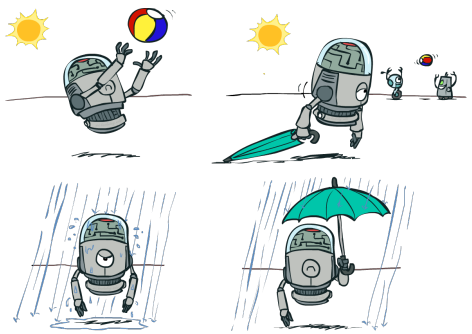
W	P(W)
sun	0.7
rain	0.3

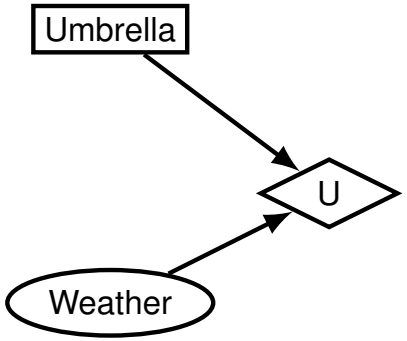
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

$$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0$$

Decision Networks





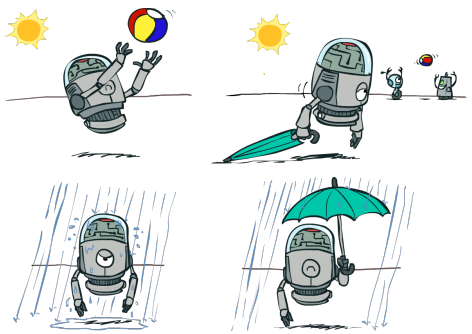
W	P(W)
sun	0.7
rain	0.3

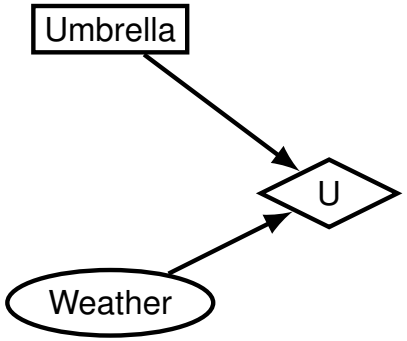
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

$$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$$

Decision Networks





W	P(W)
sun	0.7
rain	0.3

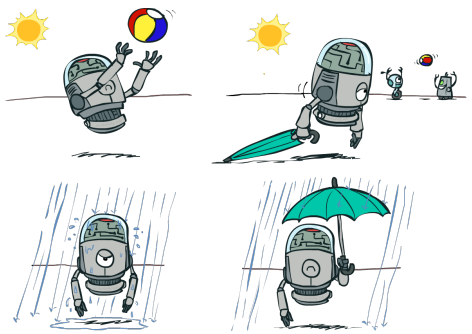
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

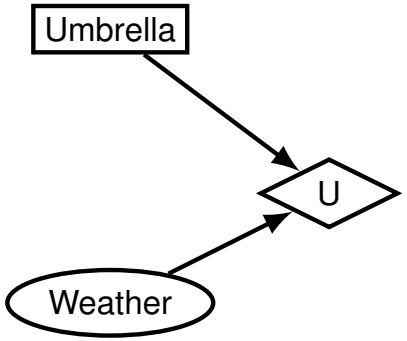
Umbrella = leave

$$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$$

Umbrella = take

Decision Networks





W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

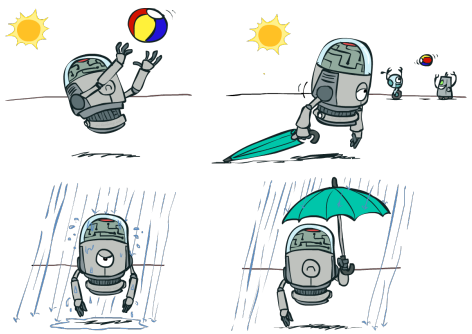
Umbrella = leave

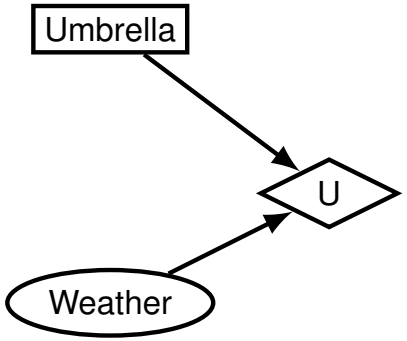
$$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$$

Umbrella = take

$$EU(take) = \sum_w P(w)U(take, w)$$

Decision Networks





W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

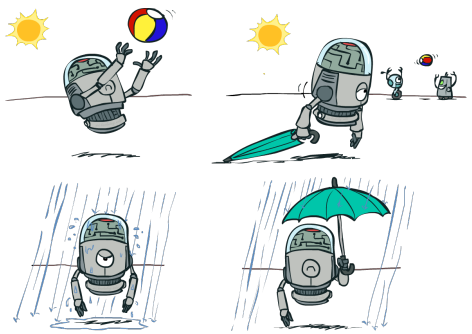
Umbrella = leave

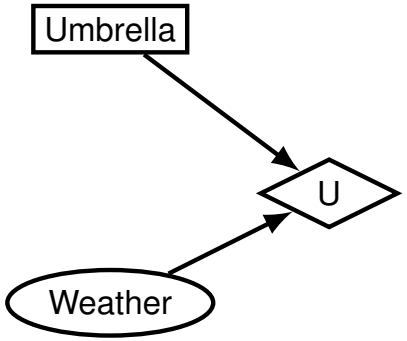
$$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$$

Umbrella = take

$$EU(take) = \sum_w P(w)U(take, w) = 0.7 \times 20 + 0.3 \times 70 = 35$$

Decision Networks





W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

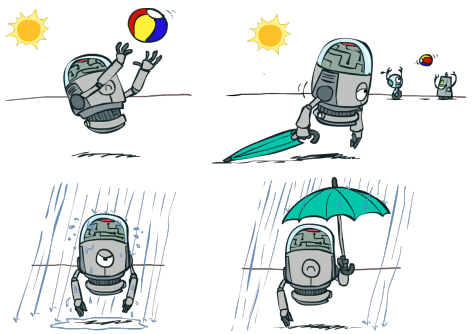
$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$

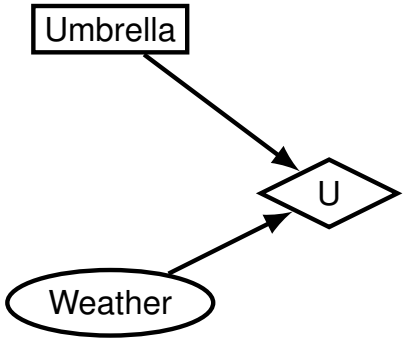
Umbrella = take

$EU(take) = \sum_w P(w)U(take, w) = 0.7 \times 20 + 0.3 \times 70 = 35$

Optimal decision = leave

Decision Networks





W	P(W)
sun	0.7
rain	0.3

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

Umbrella = leave

$EU(leave) = \sum_w P(w)U(leave, w) = 0.7 \times 100 + 0.3 \times 0 = 70$

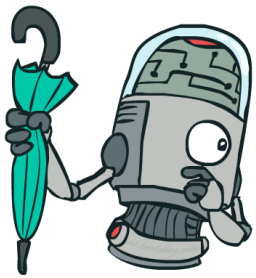
Umbrella = take

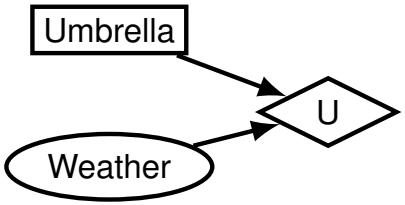
$EU(take) = \sum_w P(w)U(take, w) = 0.7 \times 20 + 0.3 \times 70 = 35$

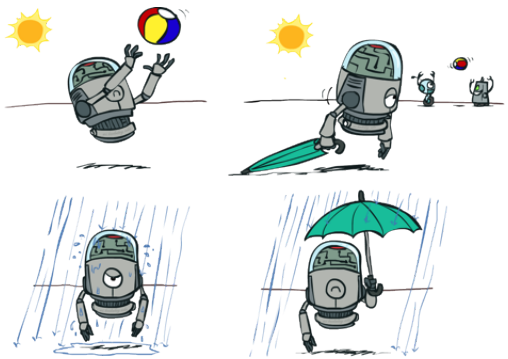
Optimal decision = leave

$MEU(\phi) == \max_a EU(a).$

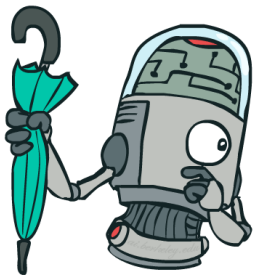
Decisions as Outcome Trees

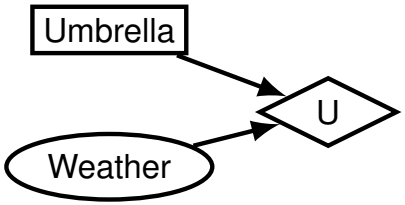


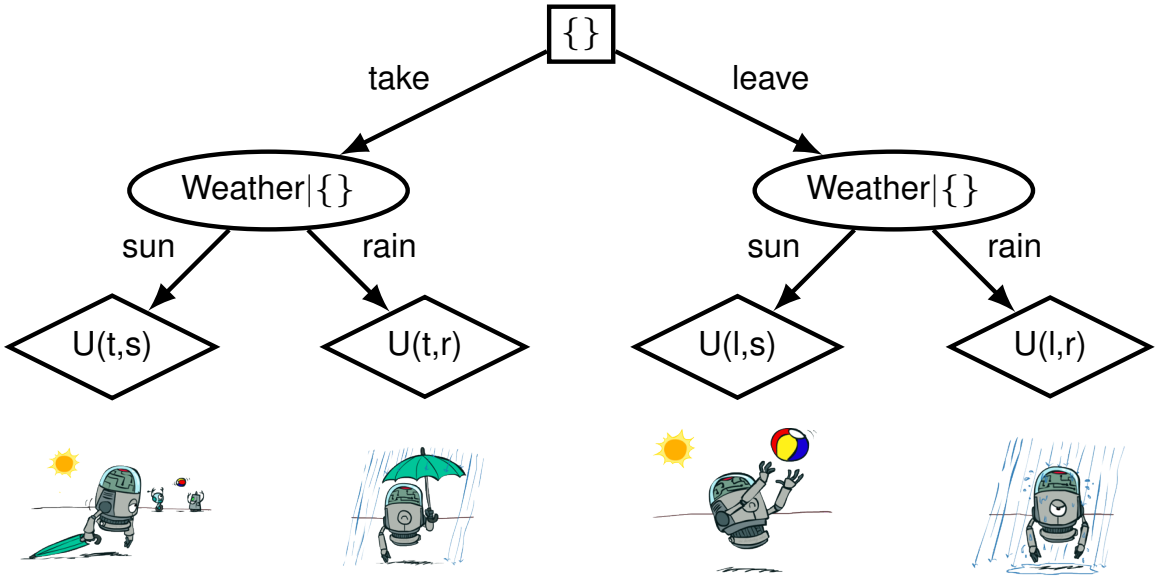


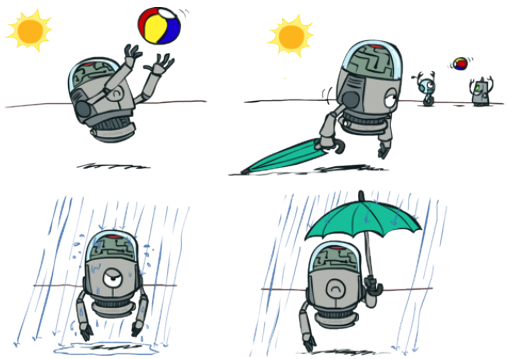


Decisions as Outcome Trees

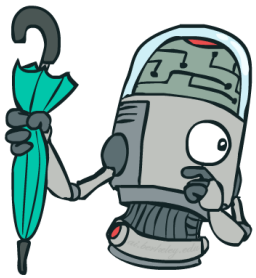


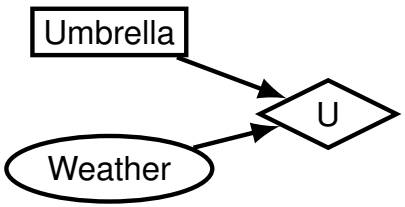


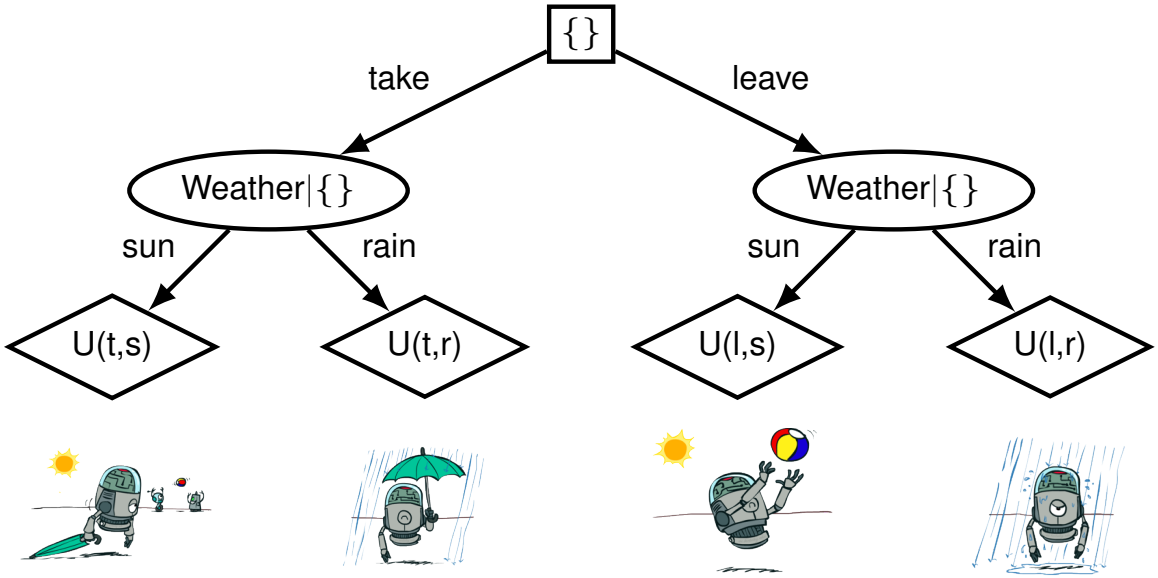


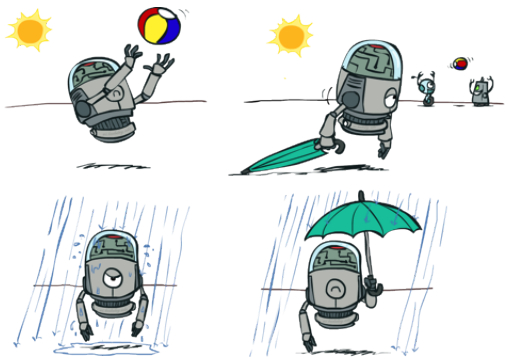


Decisions as Outcome Trees



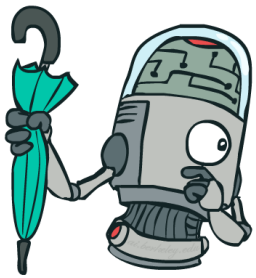


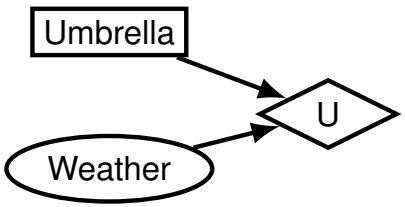


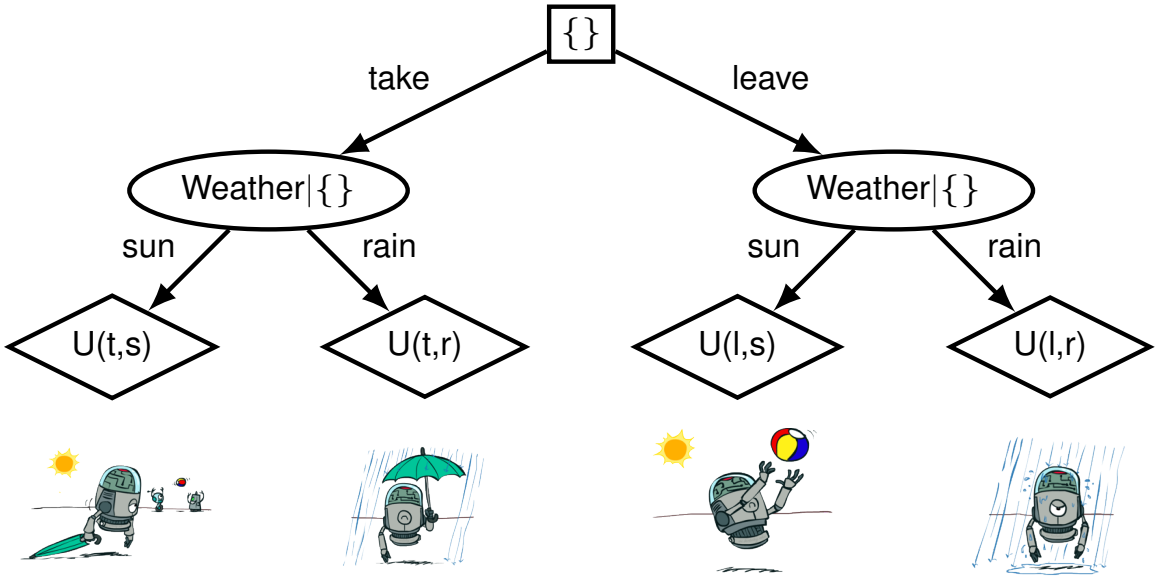


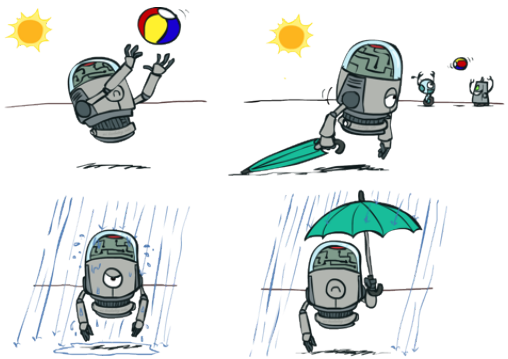
Almost exactly like expectimax / MDPs

Decisions as Outcome Trees





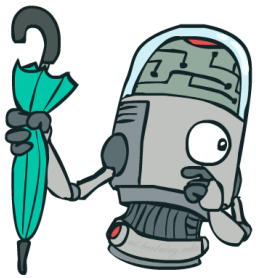


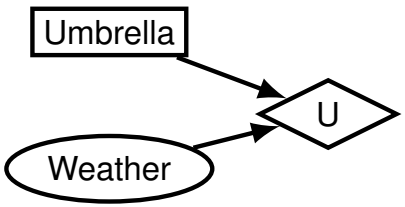


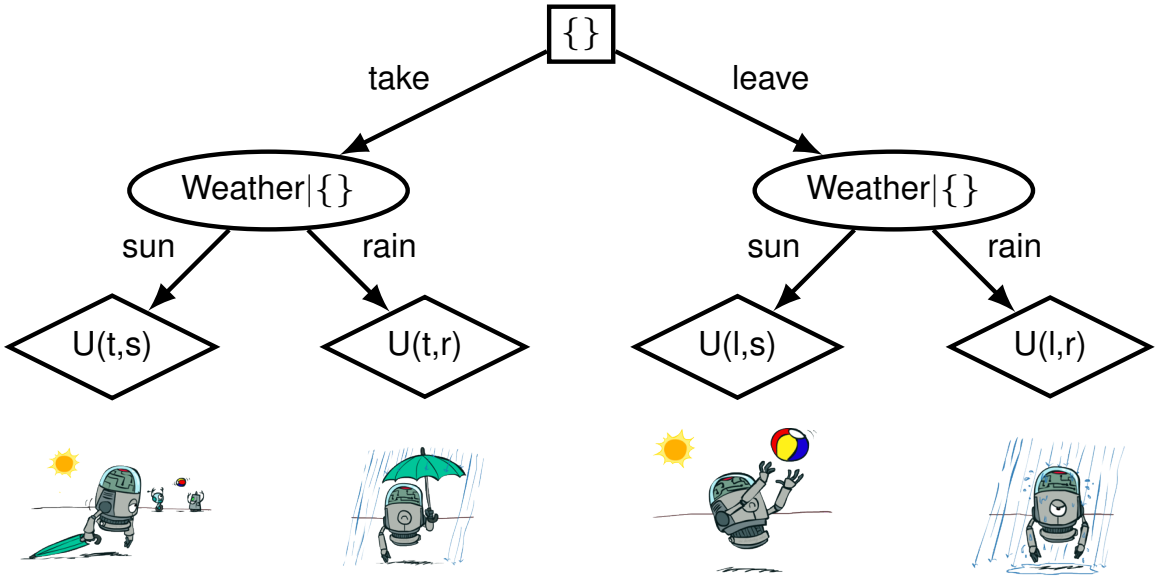
Almost exactly like expectimax / MDPs

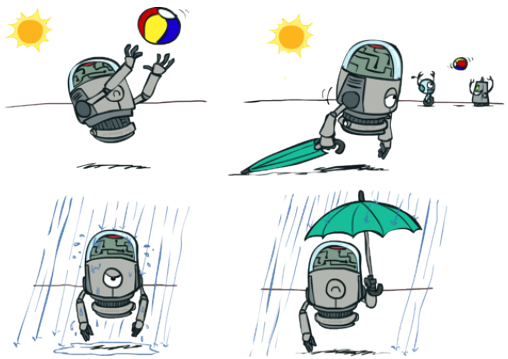
What's changed?

Decisions as Outcome Trees







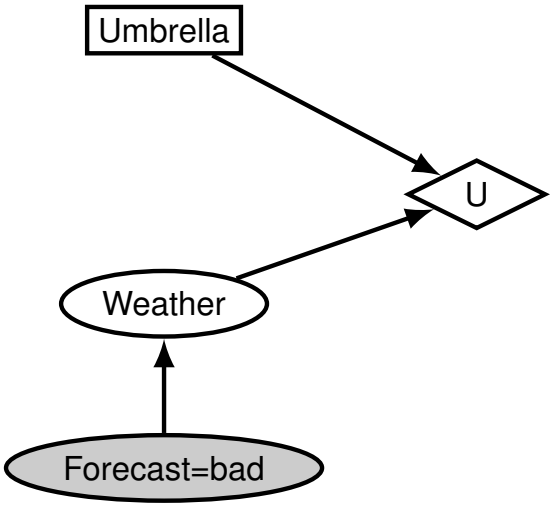


Almost exactly like expectimax / MDPs

What's changed?

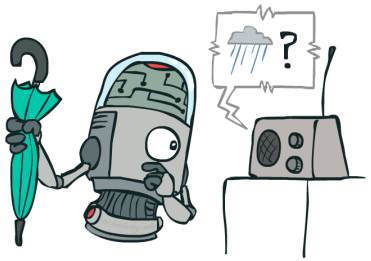
XXX: Complicated "path" states. One time reward.

Example: Decision Networks

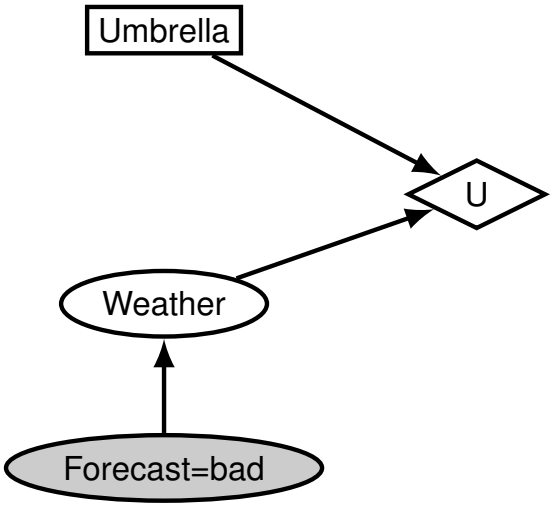


W	$P(W \mid F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



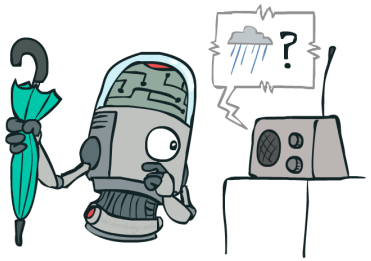
Example: Decision Networks



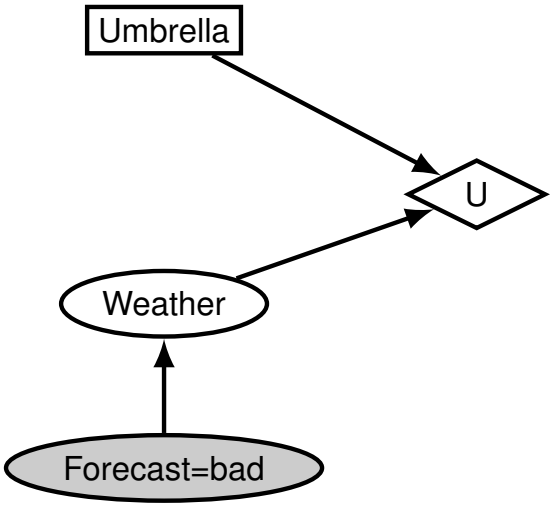
Umbrella = leave

W	$P(W\|F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

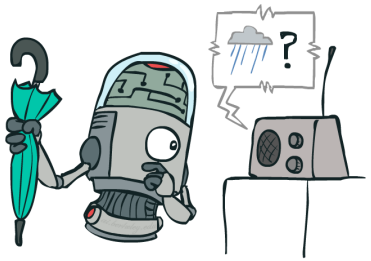


Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

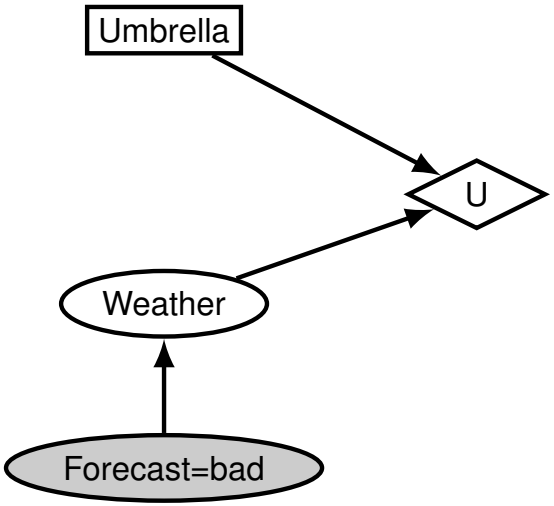
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



Umbrella = leave

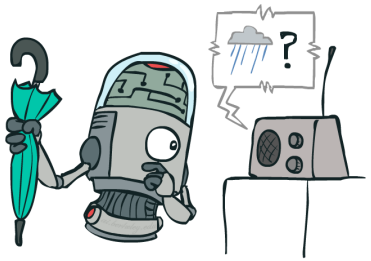
$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

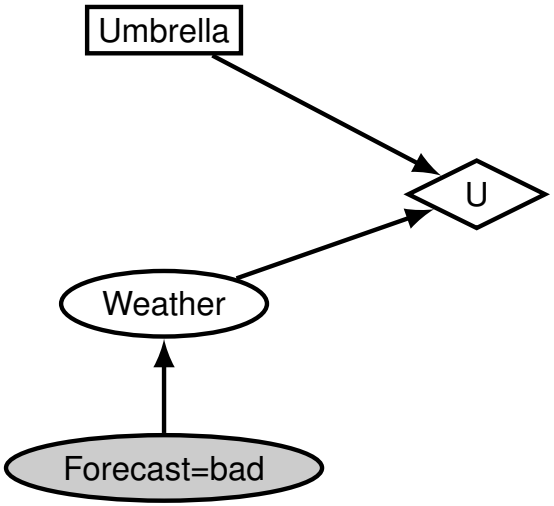
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



Umbrella = leave

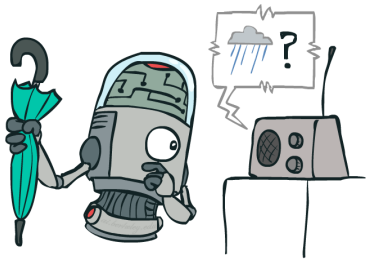
$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0$$

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

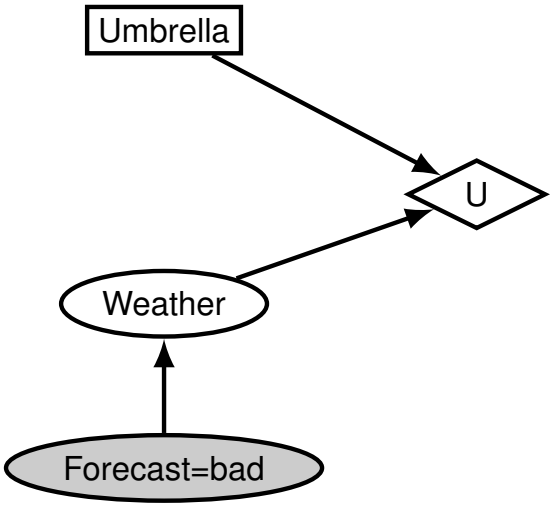
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



Umbrella = leave

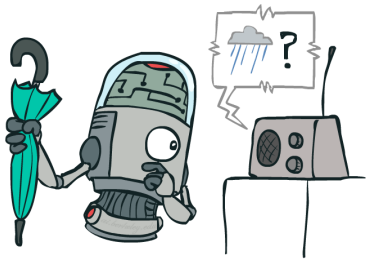
$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

Example: Decision Networks



W	$P(W \mid F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

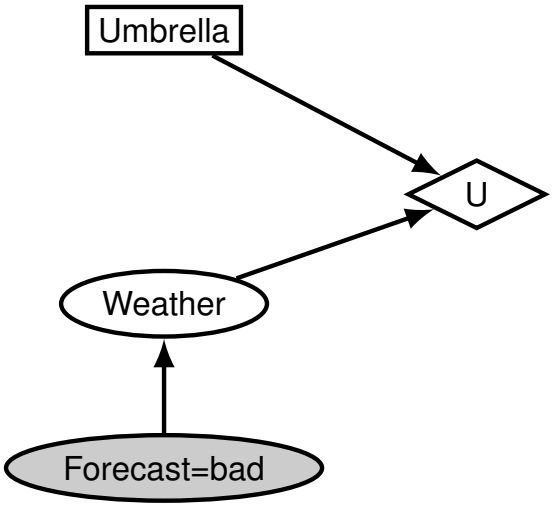


Umbrella = leave

$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

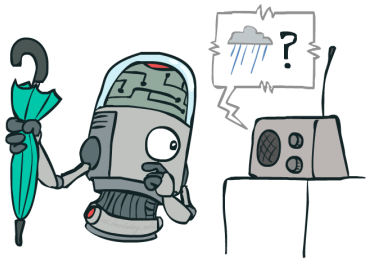
Umbrella = take

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



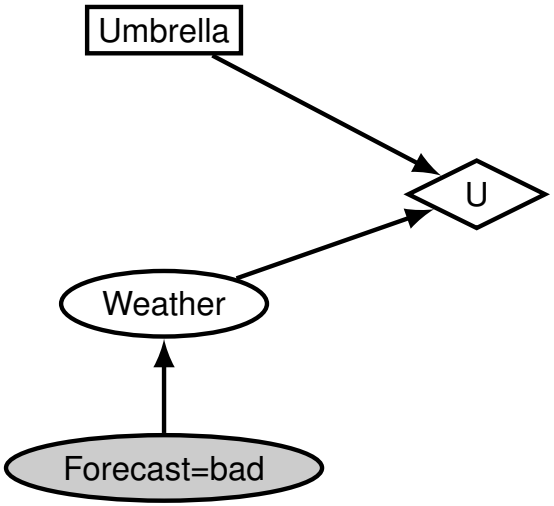
Umbrella = leave

$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

Umbrella = take

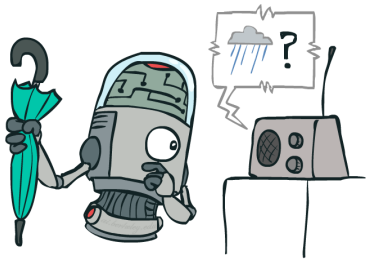
$$EU(take|bad) = \sum_w Pr(w|bad)U(take, w)$$

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



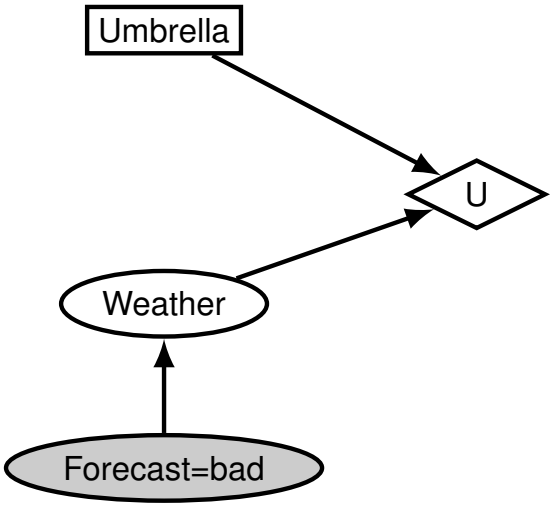
Umbrella = leave

$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

Umbrella = take

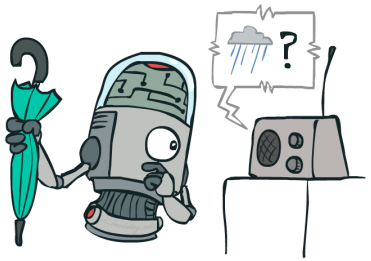
$$EU(take|bad) = \sum_w Pr(w|bad)U(take, w) = 0.34 \cdot 20 + 0.66 \cdot 70$$

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



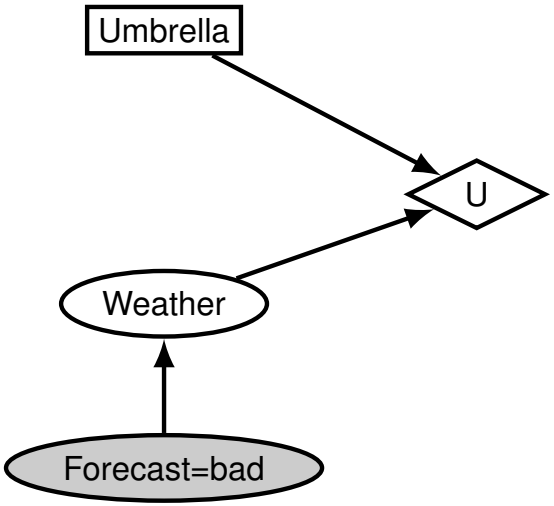
Umbrella = leave

$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

Umbrella = take

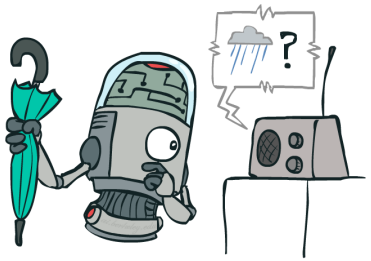
$$EU(take|bad) = \sum_w Pr(w|bad)U(take, w) = 0.34 \cdot 20 + 0.66 \cdot 70 = 53.$$

Example: Decision Networks



W	$P(W \mid F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



Umbrella = leave

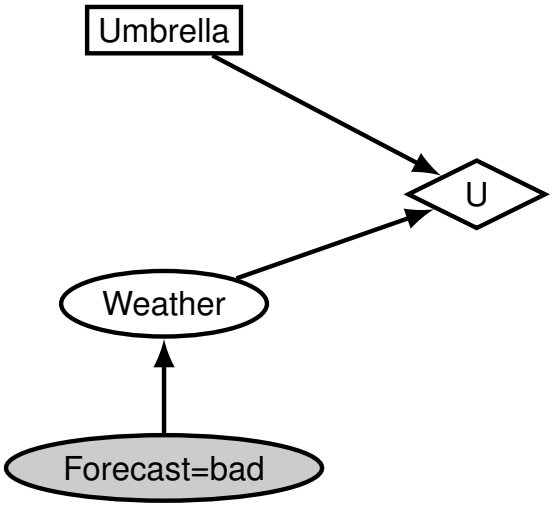
$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

Umbrella = take

$$EU(take|bad) = \sum_w Pr(w|bad)U(take, w) = 0.34 \cdot 20 + 0.66 \cdot 70 = 53.$$

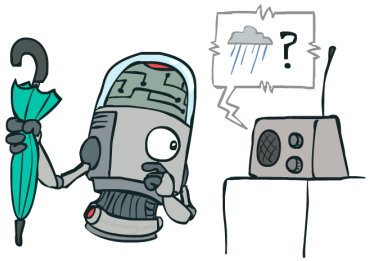
Optimal decision = take

Example: Decision Networks



W	$P(W\|F = bad)$
sun	0.34
rain	0.66

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70



Umbrella = leave

$$EU(leave|bad) = \sum_w Pr(w|bad)U(leave, w)$$
$$= 0.34 \cdot 100 + 0.66 \cdot 0 = 34.$$

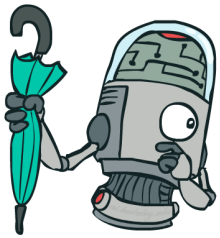
Umbrella = take

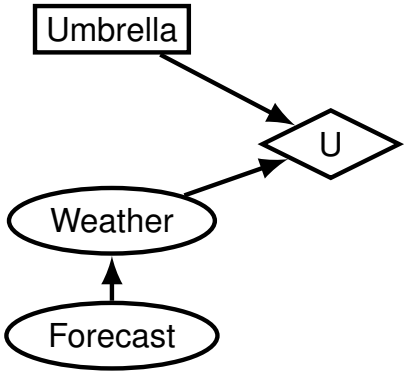
$$EU(take|bad) = \sum_w Pr(w|bad)U(take, w) = 0.34 \cdot 20 + 0.66 \cdot 70 = 53.$$

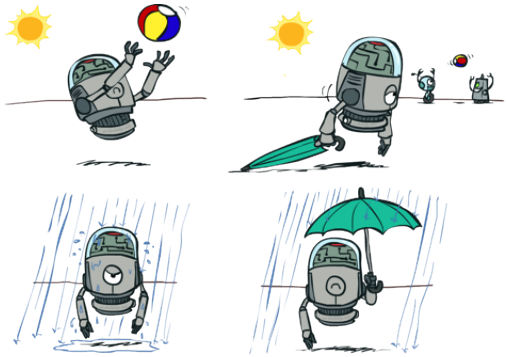
Optimal decision = take

$$MEU(F = bad) = \max_a EU(a|bad) = 53.$$

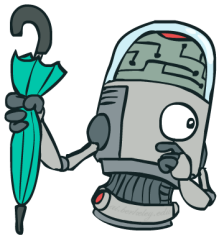
Decisions as Outcome Trees

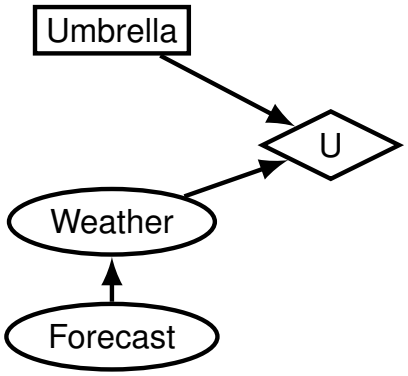


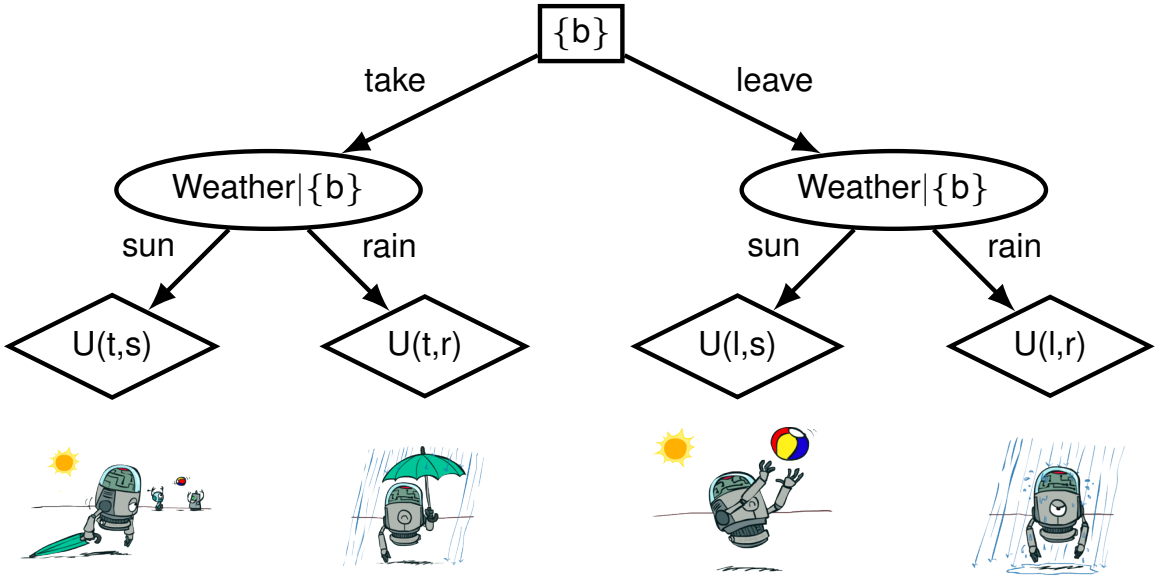


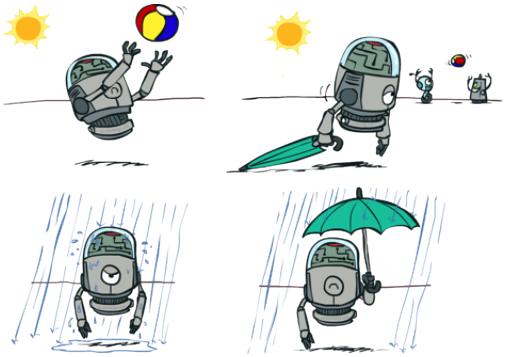


Decisions as Outcome Trees

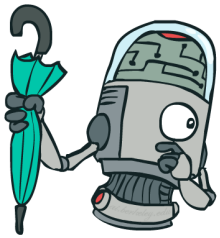


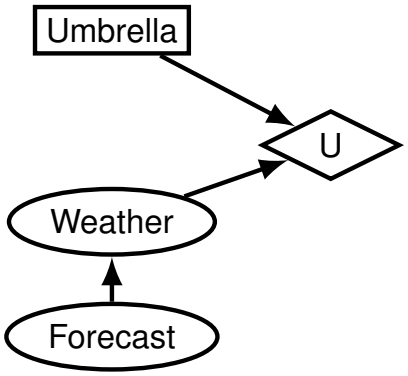


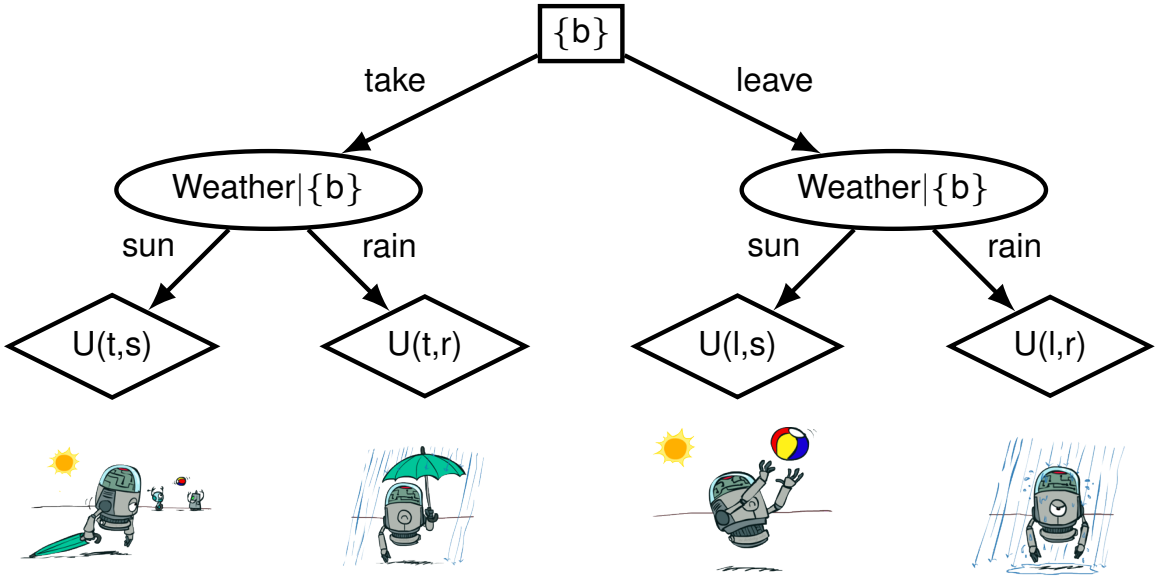


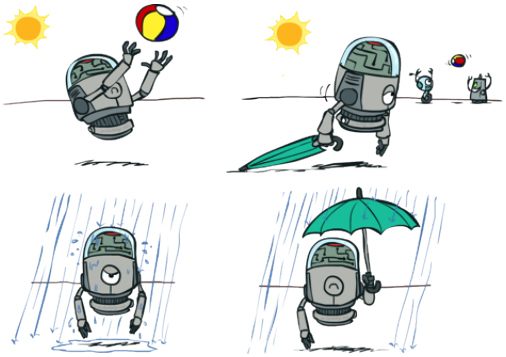


Decisions as Outcome Trees



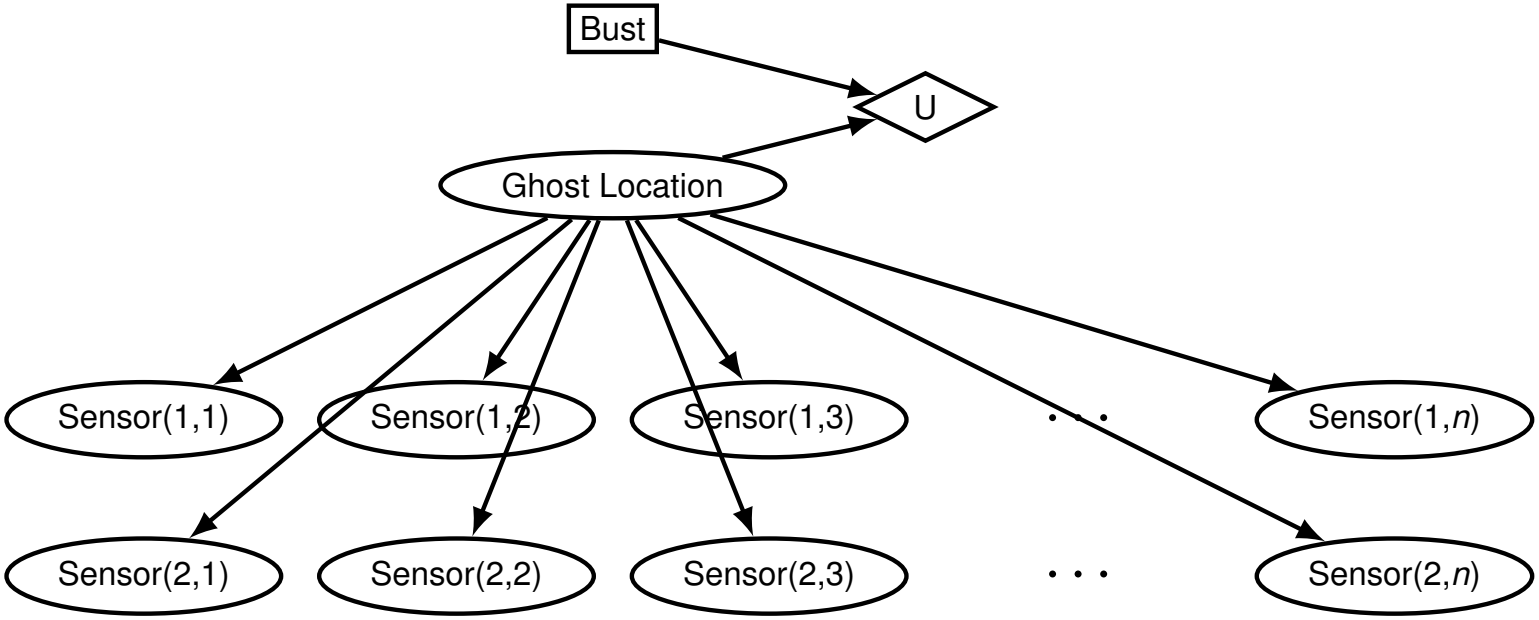




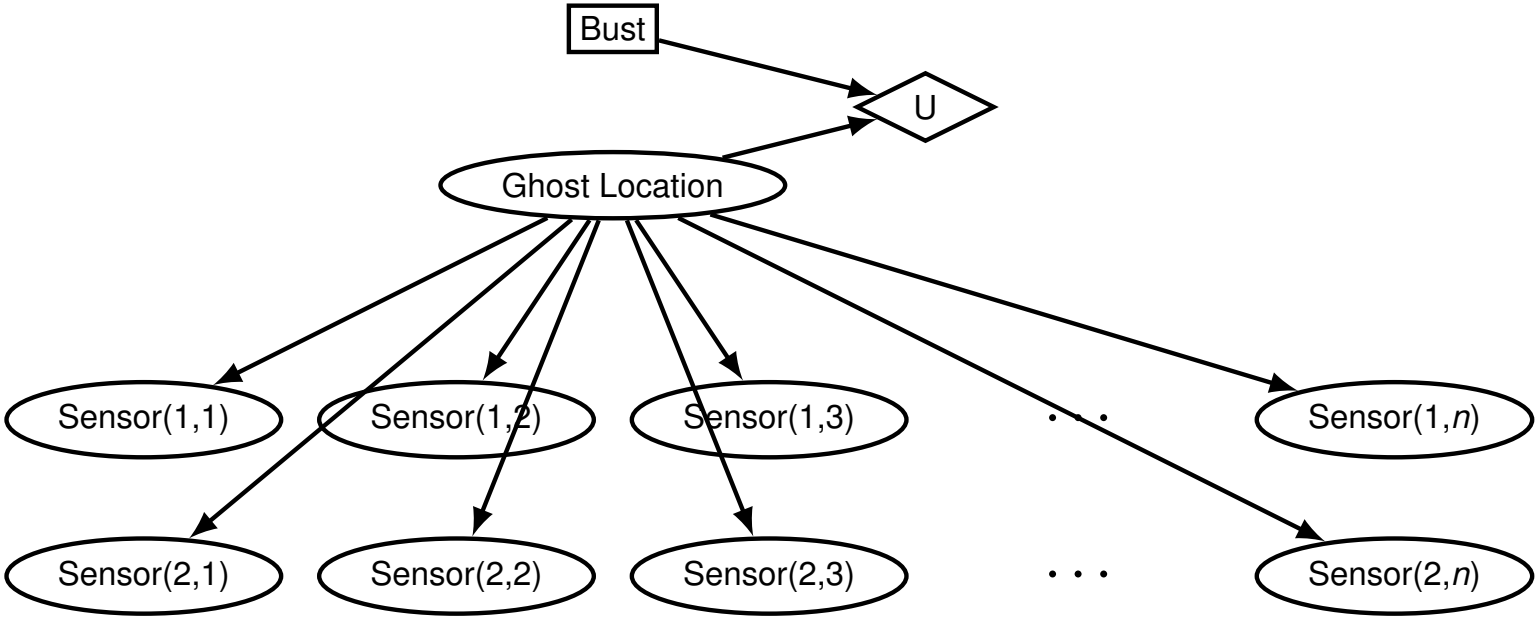


Same as before, except now we have evidence!

Ghostbusters Decision Network

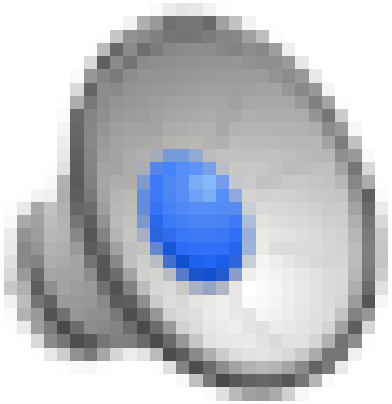


Ghostbusters Decision Network

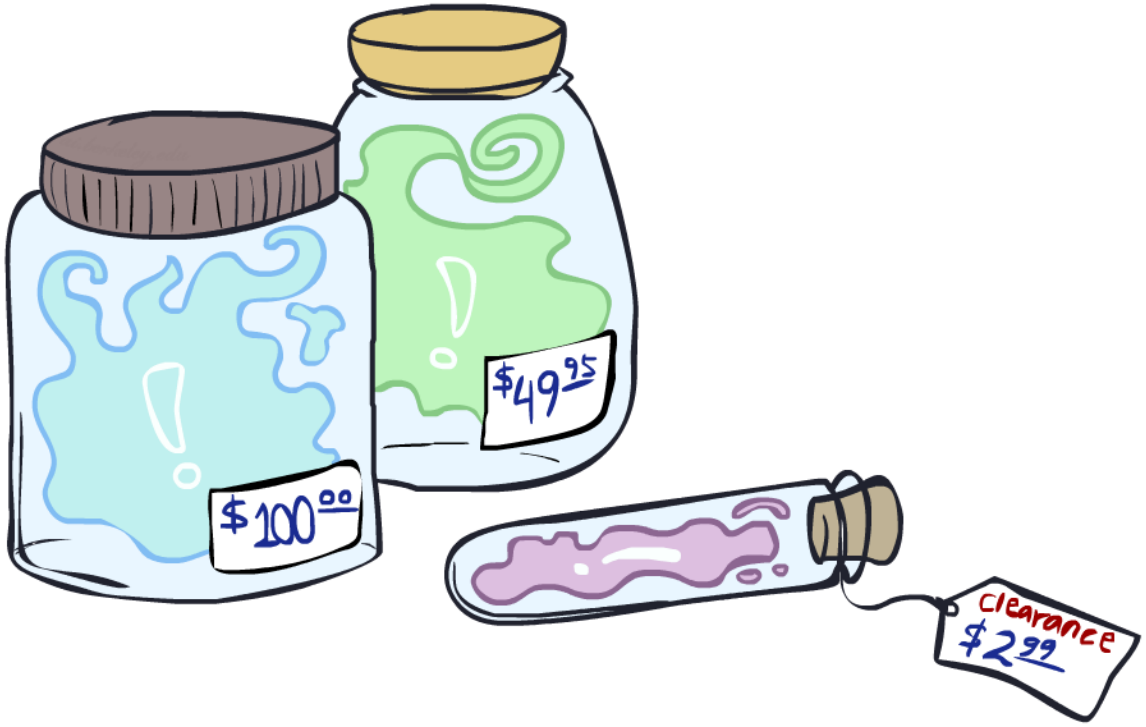


Demo: Ghostbusters with probability

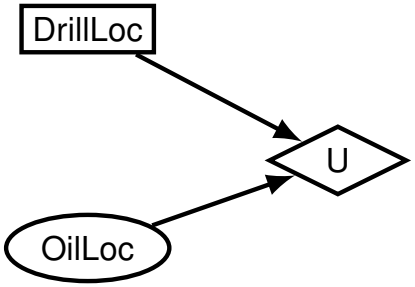
Video of Demo Ghostbusters with Probability



Value of Information

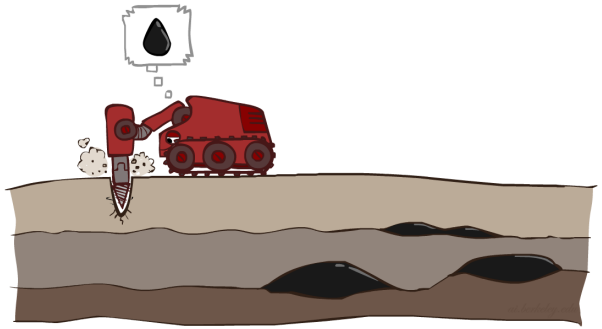


Value of Information

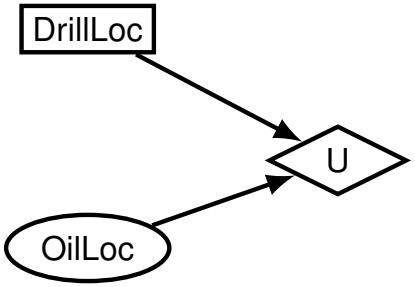


D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2

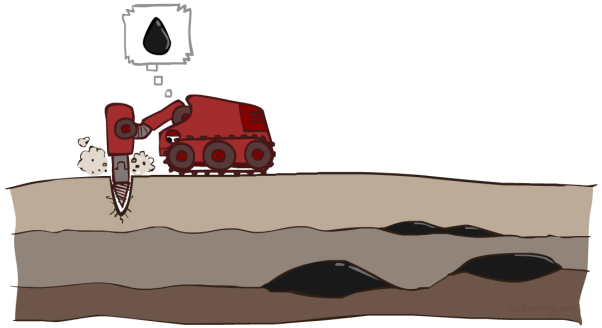


Value of Information



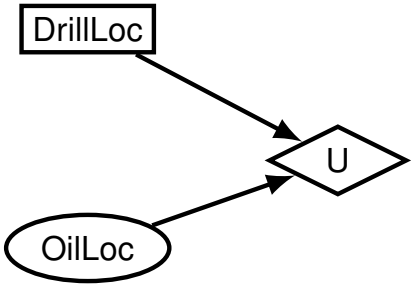
D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



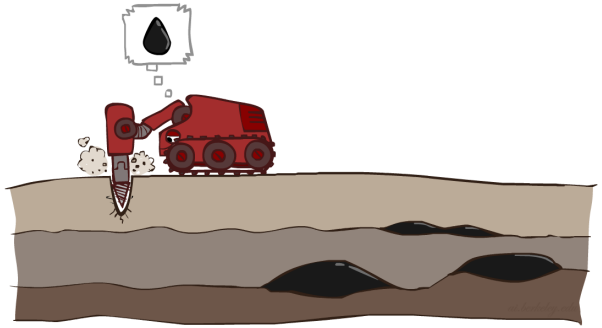
Idea: compute value of acquiring evidence

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

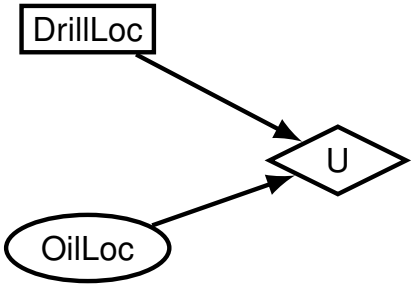
O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

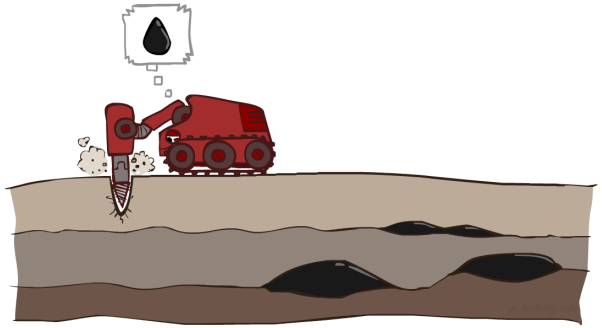
- Can be done directly from decision network

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

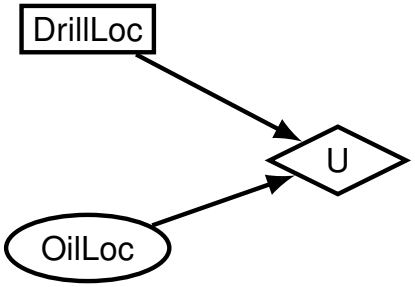
O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

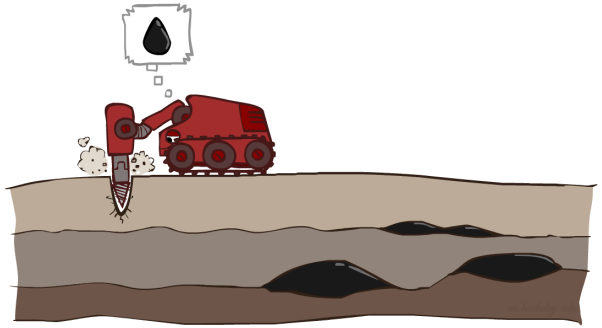
- Can be done directly from decision network

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2

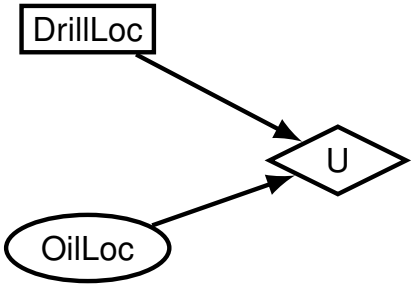


Idea: compute value of acquiring evidence

- Can be done directly from decision network

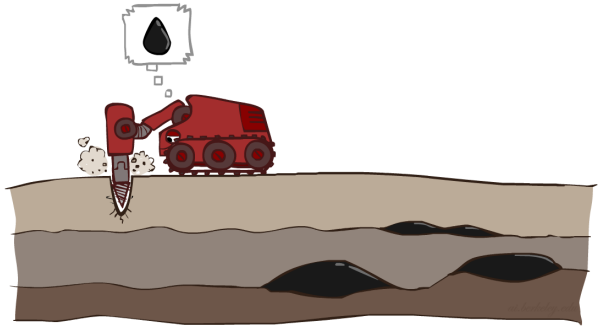
Example: buying oil drilling rights

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



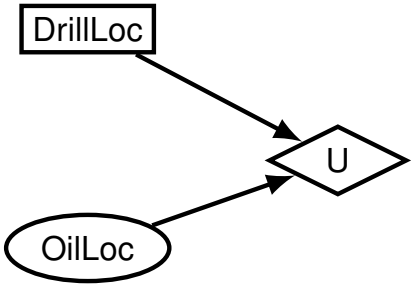
Idea: compute value of acquiring evidence

- Can be done directly from decision network

Example: buying oil drilling rights

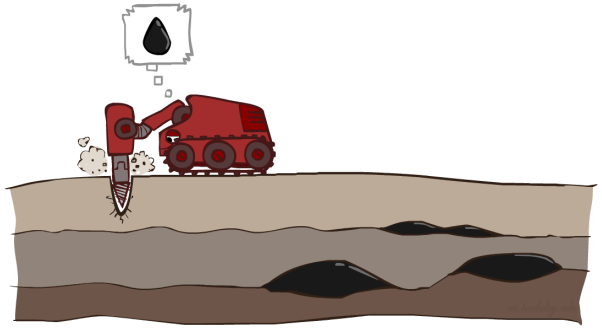
- Two blocks, oil in one of *A* or *B*.
  Equally likely.

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



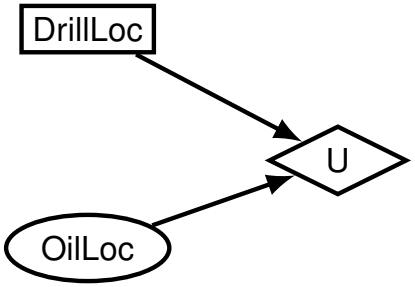
Idea: compute value of acquiring evidence

- Can be done directly from decision network

Example: buying oil drilling rights

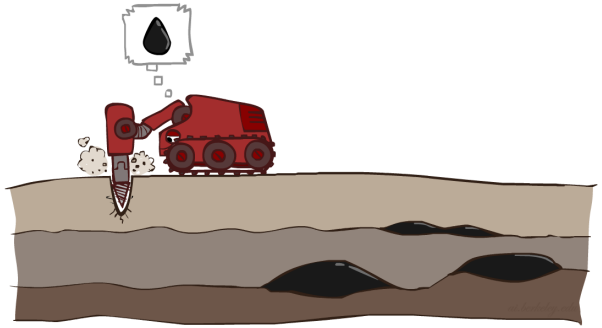
- Two blocks, oil in one of *A* or *B*. Equally likely.
- Can drill in one location

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



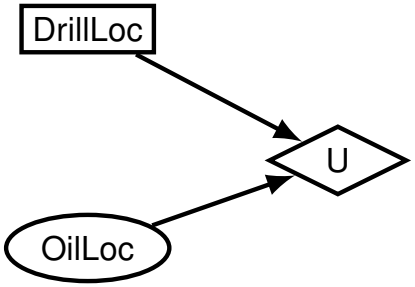
Idea: compute value of acquiring evidence

- Can be done directly from decision network

Example: buying oil drilling rights

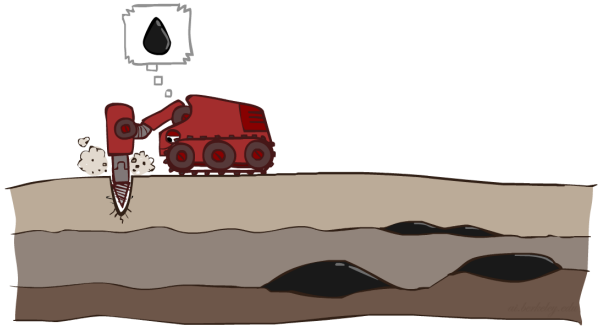
- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



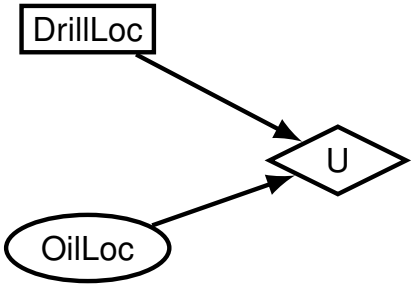
Idea: compute value of acquiring evidence

- Can be done directly from decision network

Example: buying oil drilling rights

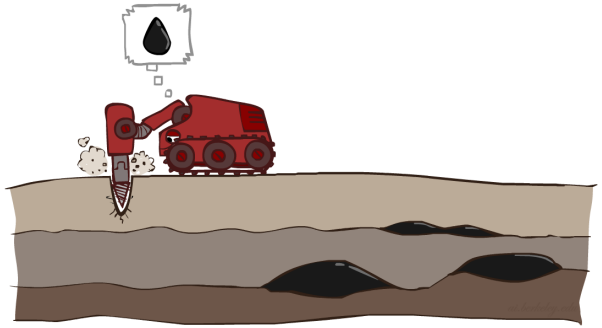
- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

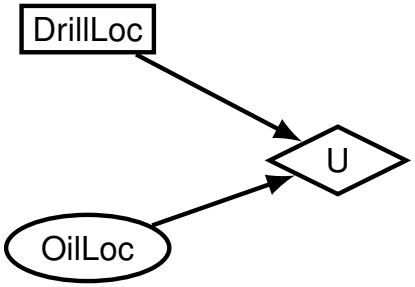
- Can be done directly from decision network

Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

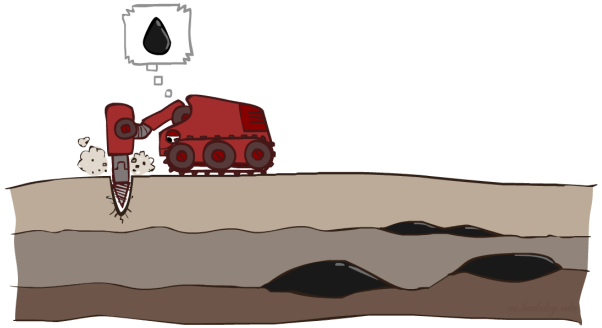
Question: what's value of information of O?

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

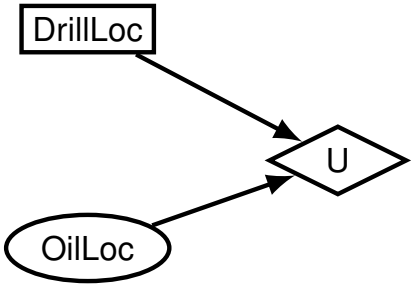
Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

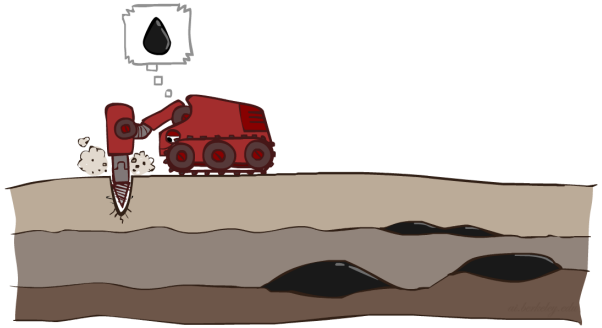
- Value of knowing whether A or B has oil.

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

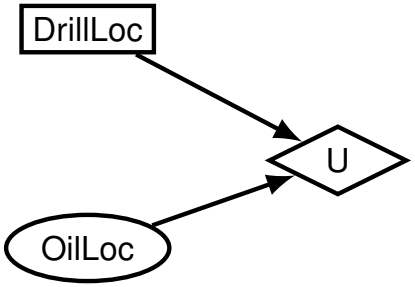
Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

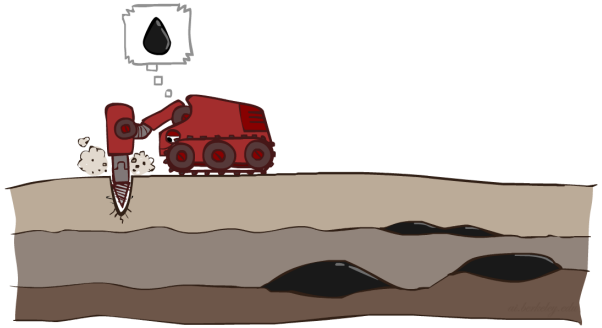
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

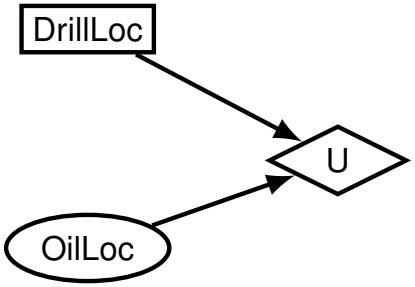
Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

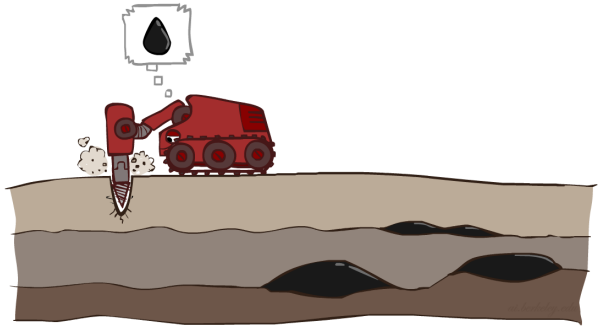
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

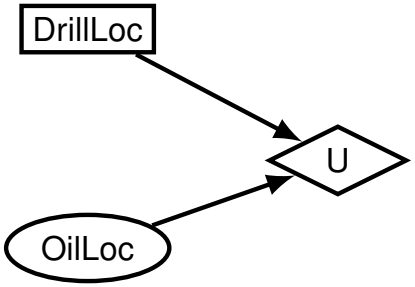
Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

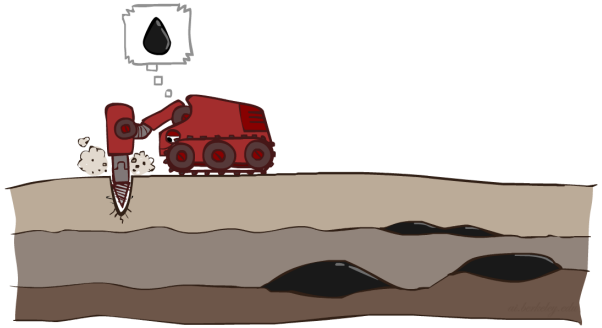
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."
- If know OilLoc, MEU is k (either way).

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

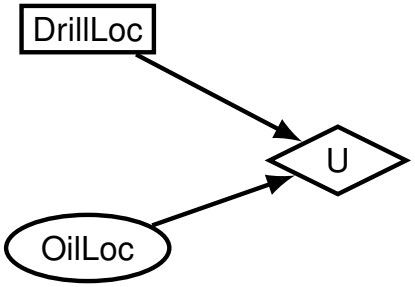
Example: buying oil drilling rights

- Two blocks, oil in one of A or B.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

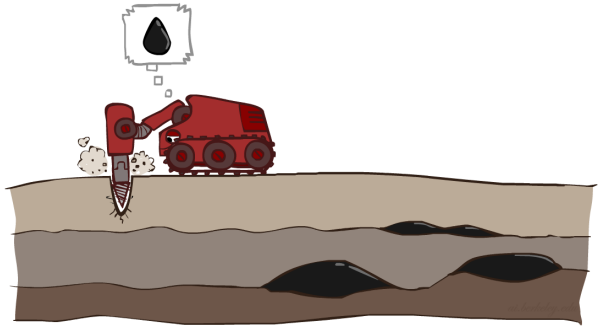
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."
- If know OilLoc, MEU is k (either way).
- Gain in MEU from knowing OilLoc?

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

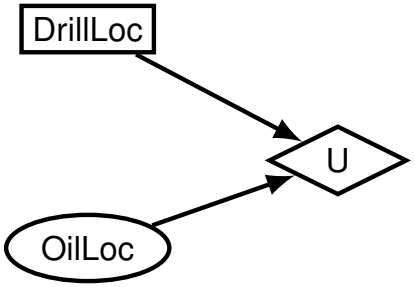
Example: buying oil drilling rights

- Two blocks, oil in one of *A* or *B*.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

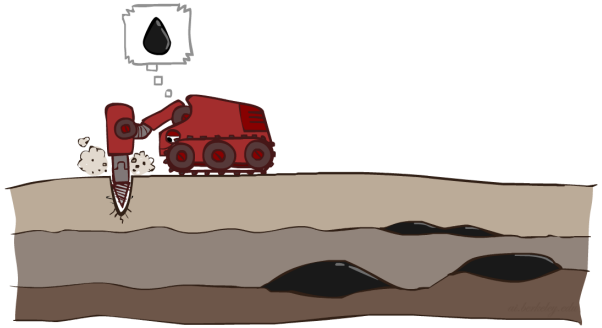
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."
- If know OilLoc, MEU is k (either way).
- Gain in MEU from knowing OilLoc?
- VPI(OilLoc) = k/2

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

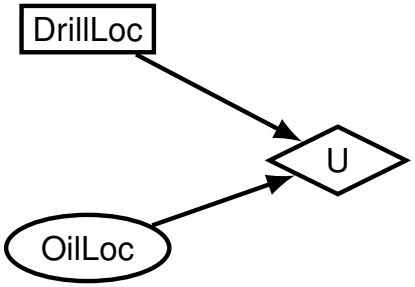
Example: buying oil drilling rights

- Two blocks, oil in one of *A* or *B*.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

Question: what's value of information of O?

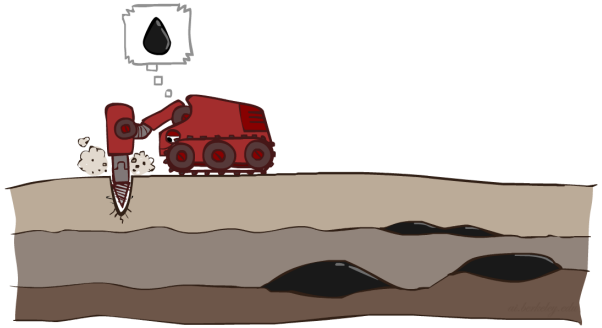
- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."
- If know OilLoc, MEU is k (either way).
- Gain in MEU from knowing OilLoc?
- VPI(OilLoc) = k/2
- Fair price of information: k/2

Value of Information



D	O	U
a	a	k
a	b	0
b	a	0
b	b	k

O	P
a	1/2
b	1/2



Idea: compute value of acquiring evidence

- Can be done directly from decision network

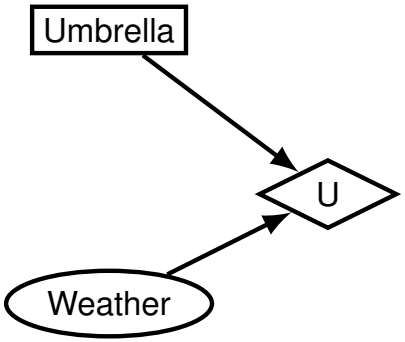
Example: buying oil drilling rights

- Two blocks, oil in one of *A* or *B*.
  Equally likely.
- Can drill in one location
- Drilling in A or B: has EU = $k/2$, MEU = $k/2$

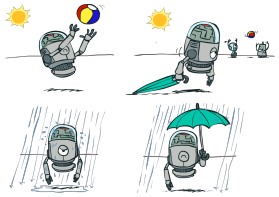
Question: what's value of information of O?

- Value of knowing whether A or B has oil.
- Value is expected gain in MEU from info.
- Survey may say "oil in a" or "oil in b."
- If know OilLoc, MEU is k (either way).
- Gain in MEU from knowing OilLoc?
- VPI(OilLoc) = k/2
- Fair price of information: k/2

VPI Example: Weather

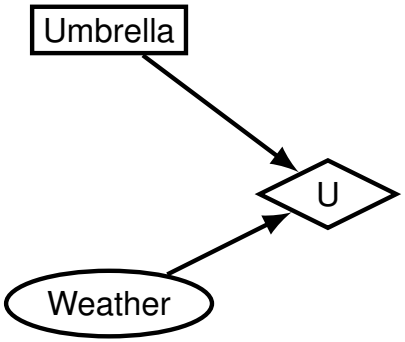


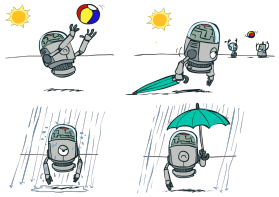
MEU with no evidence: 70



A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

VPI Example: Weather



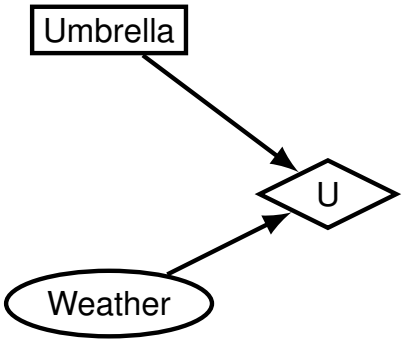


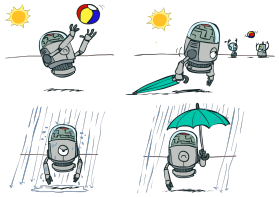
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

MEU with no evidence: 70

MEU if forecast is bad:

VPI Example: Weather



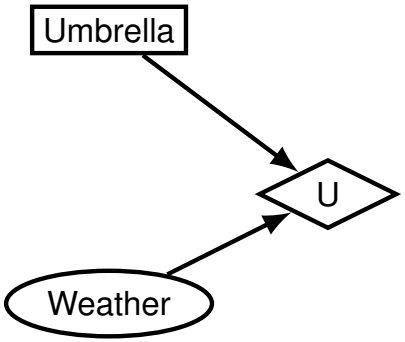


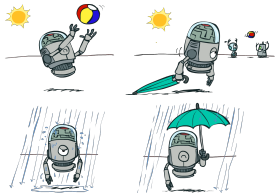
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

MEU with no evidence: 70

MEU if forecast is bad:

VPI Example: Weather





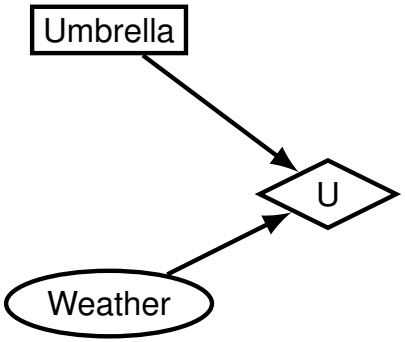
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

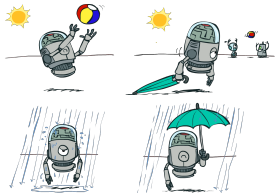
MEU with no evidence: 70

MEU if forecast is bad:

$$MEU(F = bad) = max_a EU(a|bad) = 53$$

VPI Example: Weather





A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

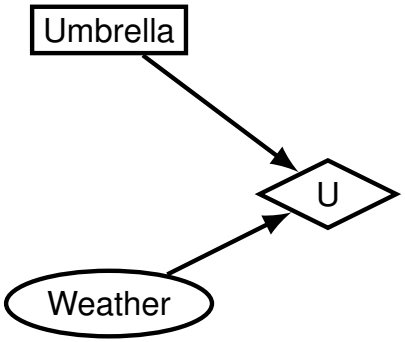
MEU with no evidence: 70

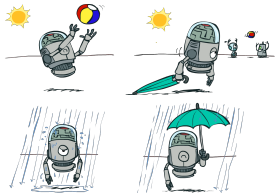
MEU if forecast is bad:

$MEU(F = bad) = max_a EU(a|bad) = 53$

MEU if forecast is good:

VPI Example: Weather





A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

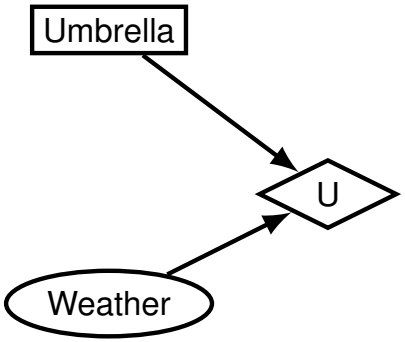
MEU with no evidence: 70

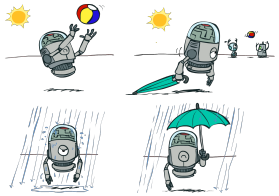
MEU if forecast is bad:

$MEU(F = bad) = max_a EU(a|bad) = 53$

MEU if forecast is good:

VPI Example: Weather





A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

MEU with no evidence: 70

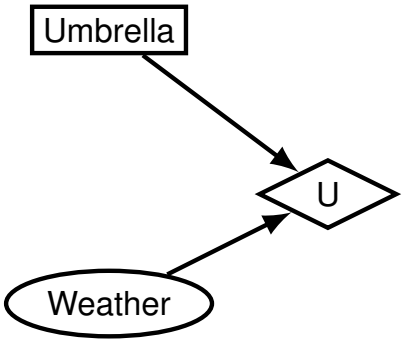
MEU if forecast is bad:

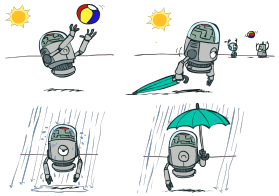
$$MEU(F = bad) = max_a EU(a|bad) = 53$$

MEU if forecast is good:

$$MEU(F = good) = max_a EU(a|good) = 95$$

VPI Example: Weather





A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

MEU with no evidence: 70

MEU if forecast is bad:

$$MEU(F = bad) = max_a EU(a|bad) = 53$$

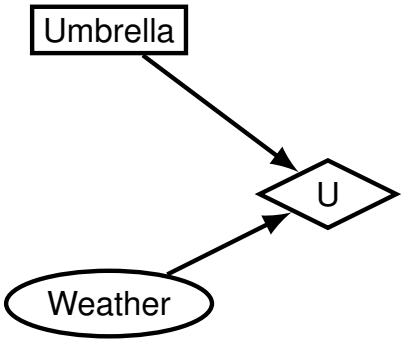
MEU if forecast is good:

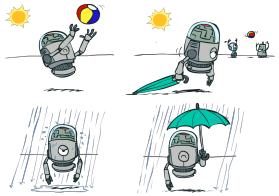
$$MEU(F = good) = max_a EU(a|good) = 95$$

Forecast distribution:

F	P(F)
good	0.59
bad	0.41

VPI Example: Weather





A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

MEU with no evidence: 70

MEU if forecast is bad:

$$MEU(F = bad) = max_a EU(a|bad) = 53$$

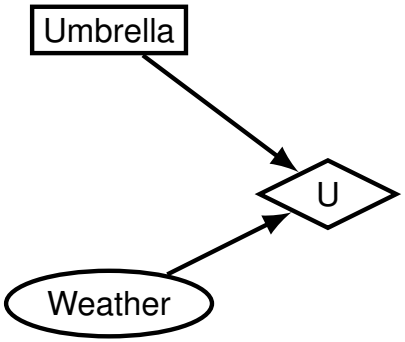
MEU if forecast is good:

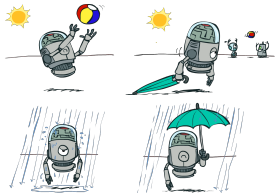
$$MEU(F = good) = max_a EU(a|good) = 95$$

Forecast distribution:

F	P(F)
good	0.59
bad	0.41

VPI Example: Weather





MEU with no evidence: 70

MEU if forecast is bad:

$MEU(F = bad) = max_a EU(a|bad) = 53$

MEU if forecast is good:

$MEU(F = good) = max_a EU(a|good) = 95$

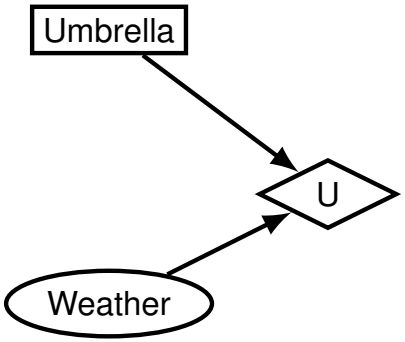
Forecast distribution:

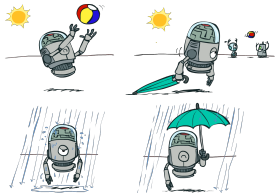
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

F	P(F)
good	0.59
bad	0.41

$$VPI(E'|e) = (\sum_{e'} P(e' \mid e) MEU(e, e')) - MEU(e)$$

VPI Example: Weather





MEU with no evidence: 70

MEU if forecast is bad:

$MEU(F = bad) = max_a EU(a|bad) = 53$

MEU if forecast is good:

$MEU(F = good) = max_a EU(a|good) = 95$

Forecast distribution:

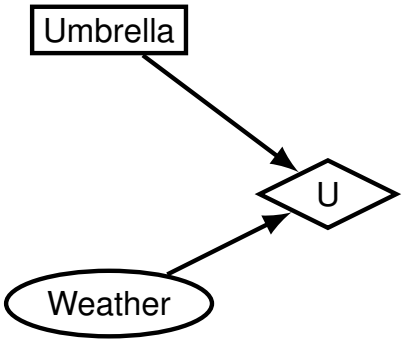
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

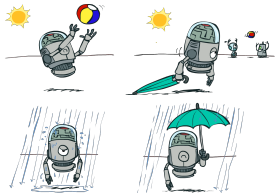
F	P(F)
good	0.59
bad	0.41

$$VPI(E'|e) = (\sum_{e'} P(e' \mid e) MEU(e, e')) - MEU(e)$$

$$VPI = 0.59 \cdot (95) + 0.41 \cdot (53) - 70$$

VPI Example: Weather





MEU with no evidence: 70

MEU if forecast is bad:

$$MEU(F = bad) = max_a EU(a|bad) = 53$$

MEU if forecast is good:

$$MEU(F = good) = max_a EU(a|good) = 95$$

Forecast distribution:

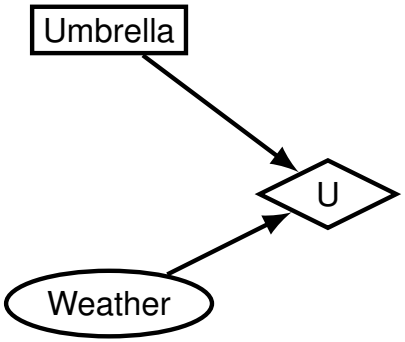
A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

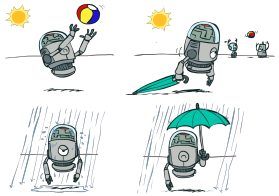
F	P(F)
good	0.59
bad	0.41

$$VPI(E'|e) = (\sum_{e'} P(e' \mid e) MEU(e, e')) - MEU(e)$$

$$VPI = 0.59 \cdot (95) + 0.41 \cdot (53) - 70 = 77.8 - 70$$

VPI Example: Weather





MEU with no evidence: 70

MEU if forecast is bad:

$MEU(F = bad) = max_a EU(a|bad) = 53$

MEU if forecast is good:

$MEU(F = good) = max_a EU(a|good) = 95$

Forecast distribution:

A	W	U(A,W)
leave	sun	100
leave	rain	0
take	sun	20
take	rain	70

F	P(F)
good	0.59
bad	0.41

$$VPI(E'|e) = (\sum_{e'} P(e' \mid e) MEU(e, e')) - MEU(e)$$

$$VPI = 0.59 \cdot (95) + 0.41 \cdot (53) - 70 = 77.8 - 70 = 7.8$$

Value of Information

Assume we have evidence $E = e$. Value if we act now:

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Assume we see that $E' = e'$. Value if we act then:

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Assume we see that $E' = e'$. Value if we act then:

$MEU(e, e') = \max_a \sum_s P(s|e, e')U(s,a)$

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Assume we see that $E' = e'$. Value if we act then:

$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$

BUT E' is a random variable whose value is unknown, so we don't know what e' will be.

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Assume we see that $E' = e'$. Value if we act then:

$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$

BUT E' is a random variable whose value is unknown, so we don't know what e' will be.

Expected value if E' is revealed and then we act:

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$

Assume we see that $E' = e'$. Value if we act then:

$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$

BUT E' is a random variable whose value is unknown, so we don't know what e' will be.

Expected value if E' is revealed and then we act:

$MEU(e,E') = \sum'_e P(e'|e)MEU(e,e')$

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$$

Assume we see that $E' = e'$. Value if we act then:

$$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$$

BUT E' is a random variable whose value is unknown, so we don't know what e' will be.

Expected value if E' is revealed and then we act:

$$MEU(e,E') = \sum_{e}' P(e'|e)MEU(e,e')$$

Value of information: how much MEU goes up by revealing E' first then acting, over acting now:

Value of Information

Assume we have evidence $E = e$. Value if we act now:

$$MEU(e) = \max_a \sum_s P(s|e)U(s,a)$$

Assume we see that $E' = e'$. Value if we act then:

$$MEU(e,e') = \max_a \sum_s P(s|e,e')U(s,a)$$

BUT E' is a random variable whose value is unknown, so we don't know what e' will be.

Expected value if E' is revealed and then we act:

$$MEU(e,E') = \sum_e' P(e'|e)MEU(e,e')$$

Value of information: how much MEU goes up by revealing E' first then acting, over acting now:

$$VPI(E'|e) = MEU(e,E') - MEU(e).$$

VPI Properties

VPI Properties

Nonnegative:

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

Nonadditive (think of observing Ej twice)

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

Nonadditive (think of observing Ej twice)

$$VPI(E_j, E_k|e) \neq VPI(E_j|e) + VPI(E_k|e)$$

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

Nonadditive (think of observing Ej twice)

$$VPI(E_j, E_k|e) \neq VPI(E_j|e) + VPI(E_k|e)$$

Order-independent

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

Nonadditive (think of observing Ej twice)

$$VPI(E_j, E_k|e) \neq VPI(E_j|e) + VPI(E_k|e)$$

Order-independent

$$VPI(E_j, E_k|e) = VPI(E_j|e) + VPI(E_k|e, E_j)$$

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

Nonadditive (think of observing Ej twice)

$$VPI(E_j, E_k|e) \neq VPI(E_j|e) + VPI(E_k|e)$$

Order-independent

$$VPI(E_j, E_k|e) = VPI(E_j|e) + VPI(E_k|e, E_j)$$

VPI Properties

Nonnegative:

$$\forall E', e : VPI(E' \mid e) \geq 0$$

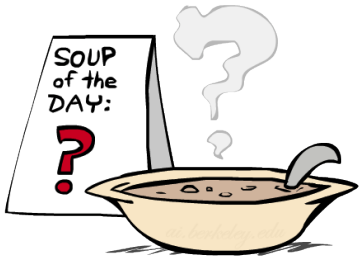
Nonadditive (think of observing Ej twice)

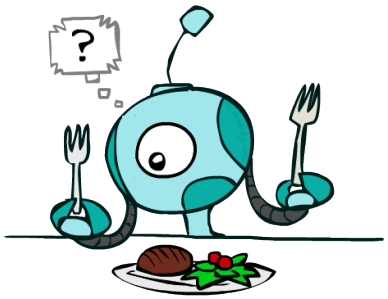
$$VPI(E_j, E_k|e) \neq VPI(E_j|e) + VPI(E_k|e)$$

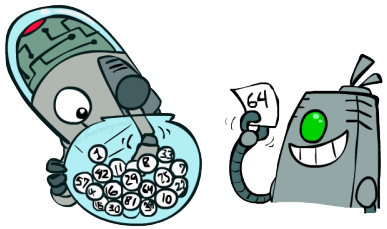
Order-independent

$$VPI(E_j, E_k|e) = VPI(E_j|e) + VPI(E_k|e, E_j)$$
$$= VPI(E_k|e) + VPI(E_j|e, E_k)$$

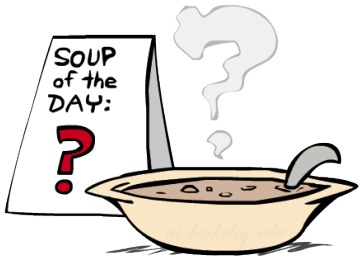
Quick VPI Questions



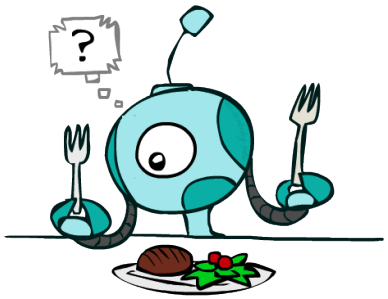


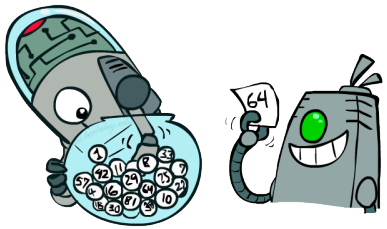


Quick VPI Questions

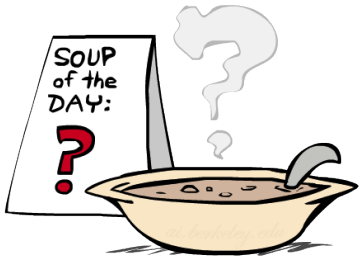


The soup of the day is either clam chowder or split pea, but you wouldn't order either one. What's the value of knowing which it is?

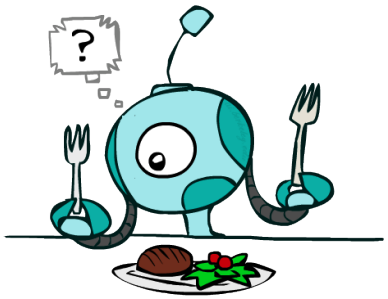




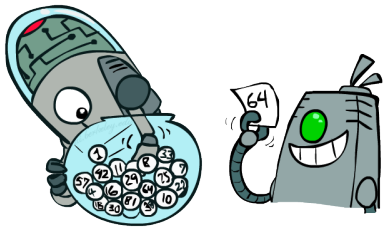
Quick VPI Questions



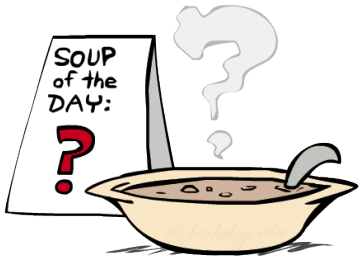
The soup of the day is either clam chowder or split pea, but you wouldn't order either one. What's the value of knowing which it is?



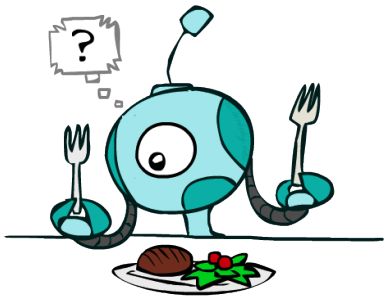
There are two kinds of plastic forks at a picnic. One kind is slightly sturdier. What's the value of knowing which?



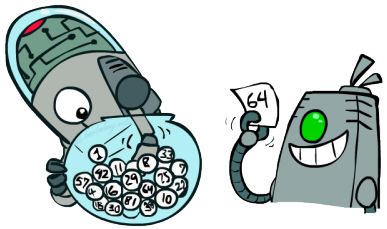
Quick VPI Questions



The soup of the day is either clam chowder or split pea, but you wouldn't order either one. What's the value of knowing which it is?



There are two kinds of plastic forks at a picnic. One kind is slightly sturdier. What's the value of knowing which?



You're playing the lottery. The prize will be $0 or $100. You can play any number between 1 and 100 (chance of winning is 1%). What is the value of knowing the winning number?

Value of Imperfect Information?

Value of Imperfect Information?

No such thing

Value of Imperfect Information?

No such thing

Information corresponds to the observation of a node in the decision network

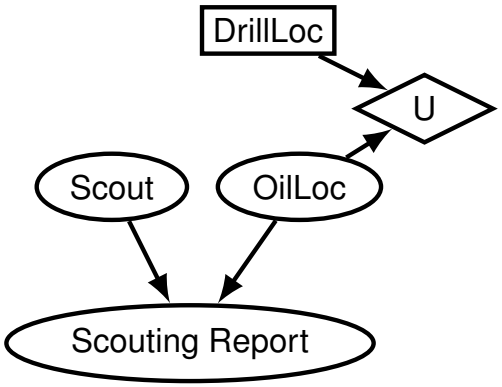
Value of Imperfect Information?

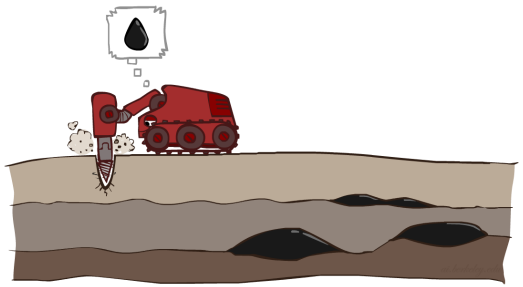
No such thing

Information corresponds to the observation of a node in the decision network

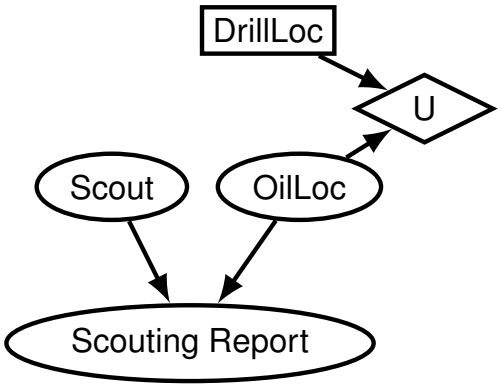
"Noisy" means don't observe another variable:
noisy version of original variable

VPI Question

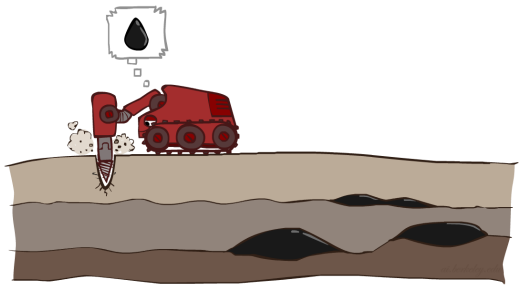




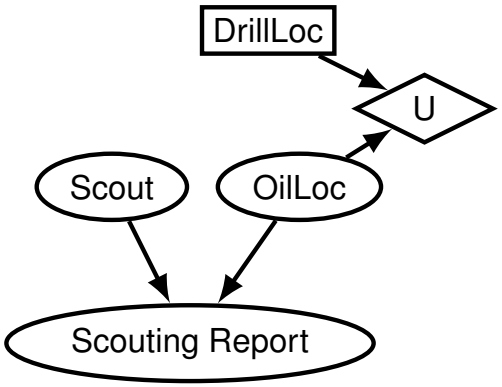
VPI Question



VPI(OilLoc) ?

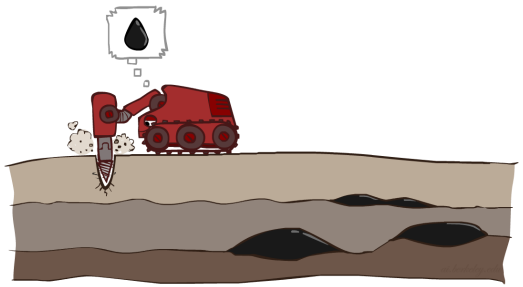


VPI Question

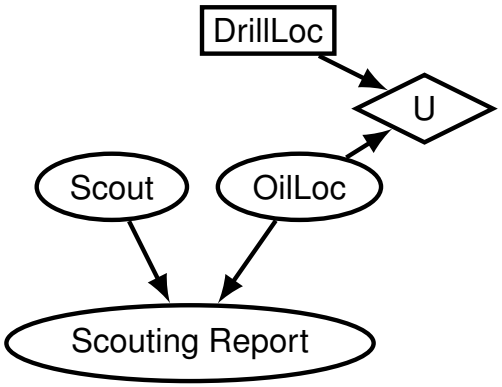


VPI(OilLoc) ?

VPI(ScoutingReport) ?



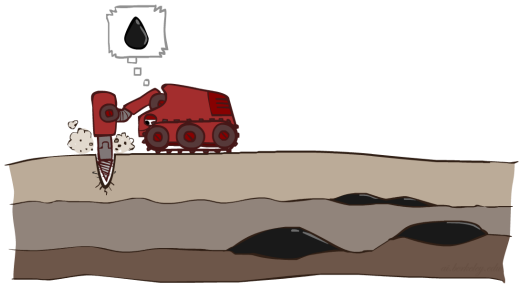
VPI Question



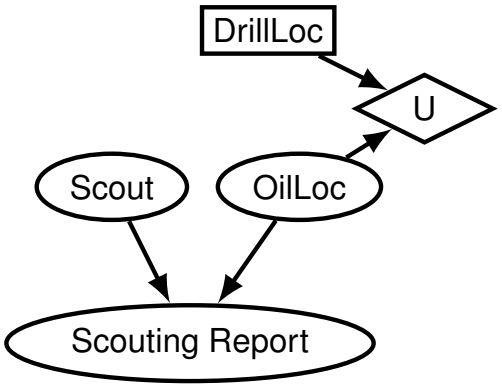
VPI(OilLoc) ?

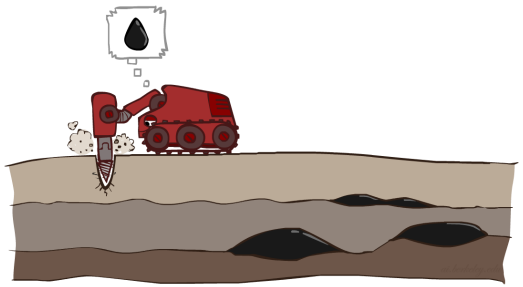
VPI(ScoutingReport) ?

VPI(Scout) ?



VPI Question





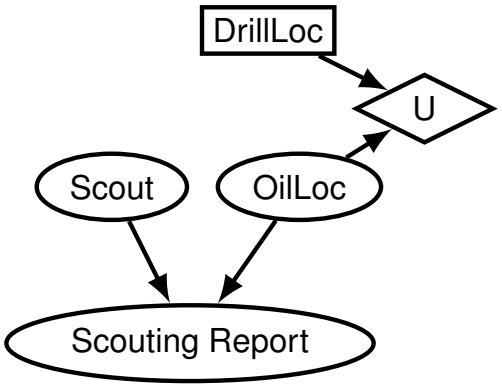
VPI(OilLoc) ?

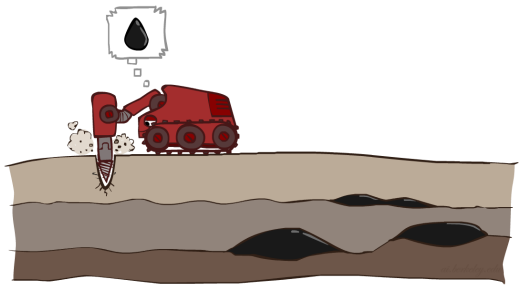
VPI(ScoutingReport) ?

VPI(Scout) ?

VPI(Scout | ScoutingReport) ?

VPI Question





VPI(OilLoc) ?

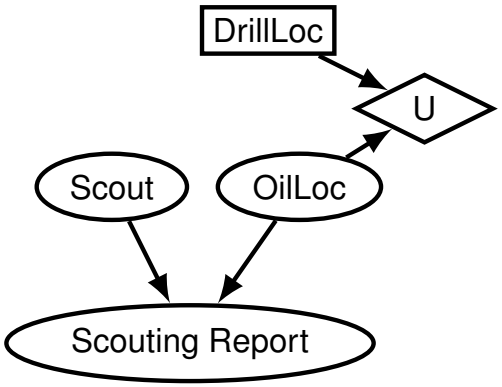
VPI(ScoutingReport) ?

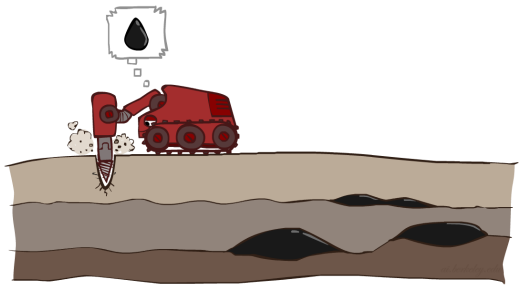
VPI(Scout) ?

VPI(Scout | ScoutingReport) ?

Generally:

VPI Question





VPI(OilLoc) ?

VPI(ScoutingReport) ?

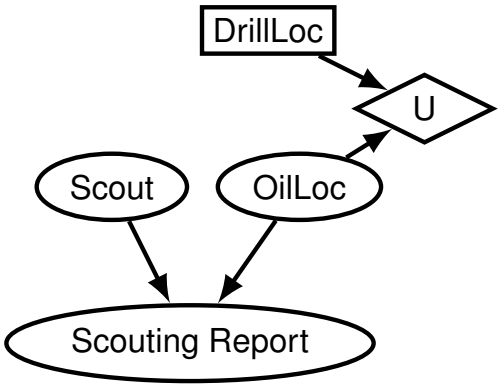
VPI(Scout) ?

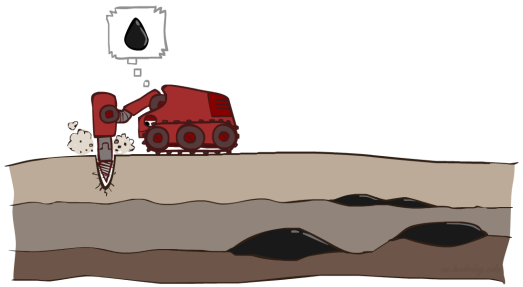
VPI(Scout | ScoutingReport) ?

Generally:

- If Parents(U) $\perp\!\!\!\perp$ Z | CurrentEvidence

VPI Question





VPI(OilLoc) ?

VPI(ScoutingReport) ?

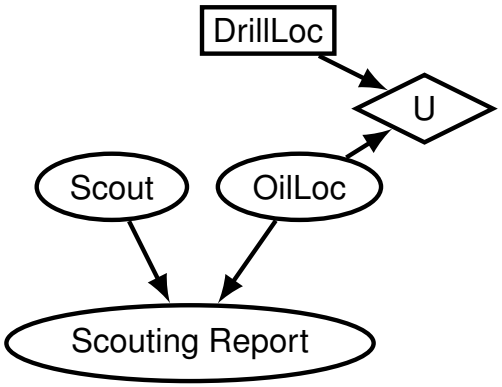
VPI(Scout) ?

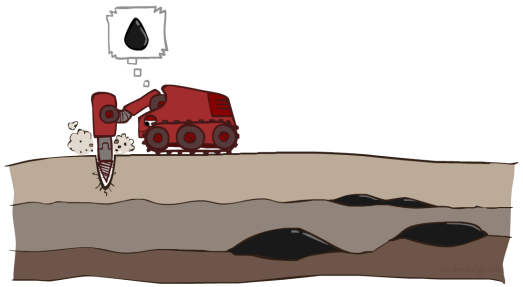
VPI(Scout | ScoutingReport) ?

Generally:

- If Parents(U) $\perp\!\!\!\perp$ Z | CurrentEvidence
- Then VPI(Z | CurrentEvidence) = 0

VPI Question





VPI(OilLoc) ?

VPI(ScoutingReport) ?

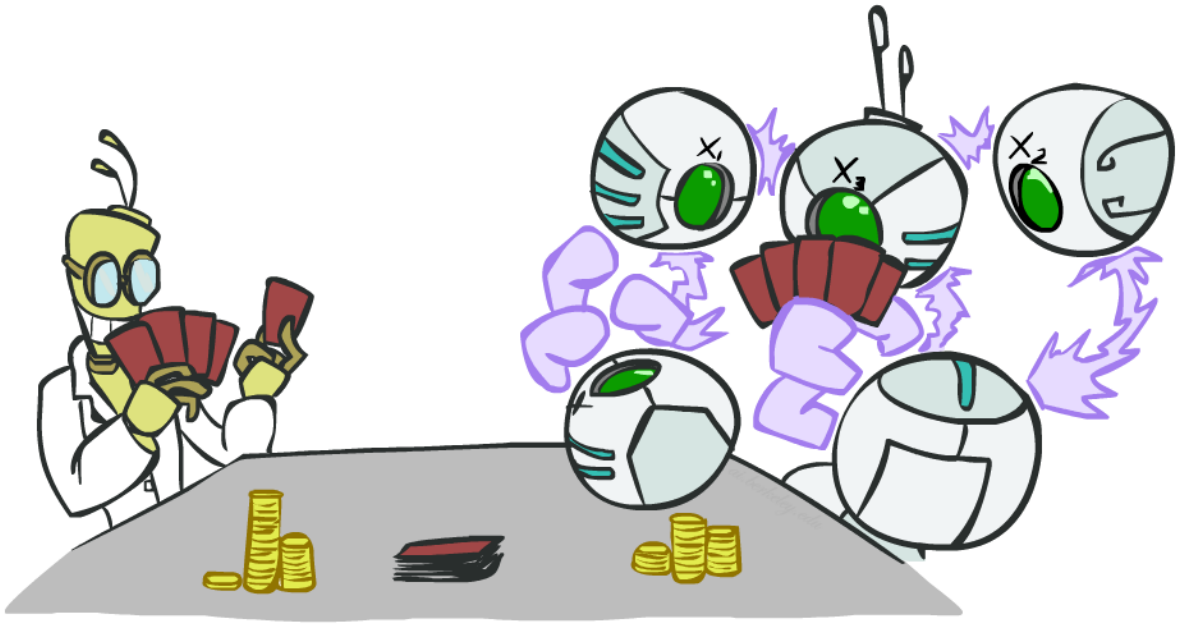
VPI(Scout) ?

VPI(Scout | ScoutingReport) ?

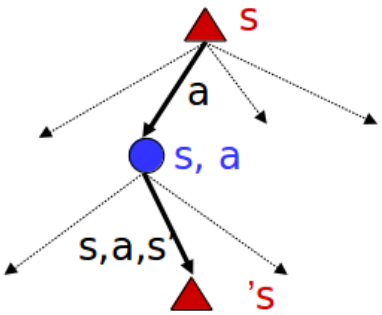
Generally:

- If Parents(U) $\perp\!\!\!\perp$ Z | CurrentEvidence
- Then VPI(Z | CurrentEvidence) = 0

POMDPs

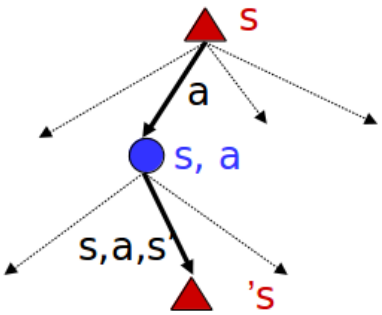


POMDPs

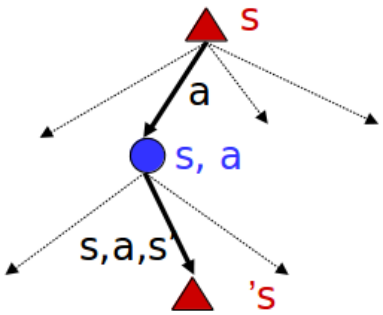


POMDPs

MDPs have:



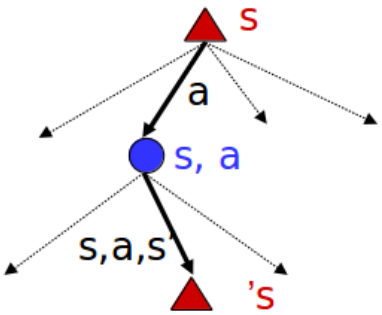
POMDPs



MDPs have:

- States S

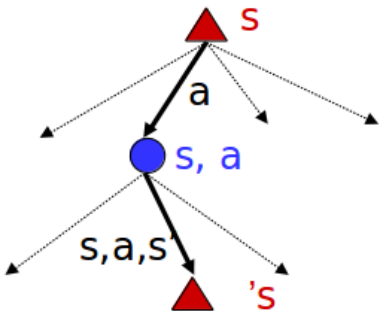
POMDPs



MDPs have:

- States S
- Actions A

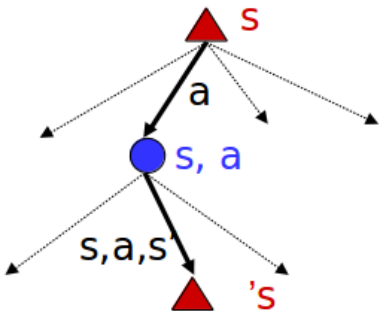
POMDPs



MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))

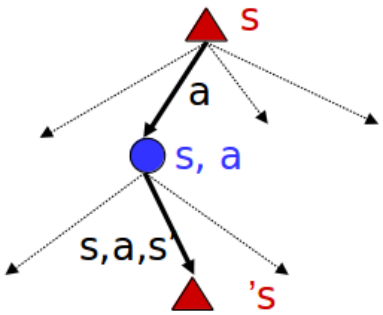
POMDPs



MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

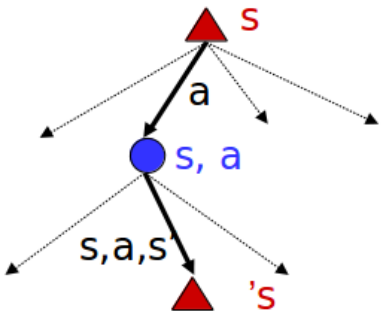
POMDPs



MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

POMDPs

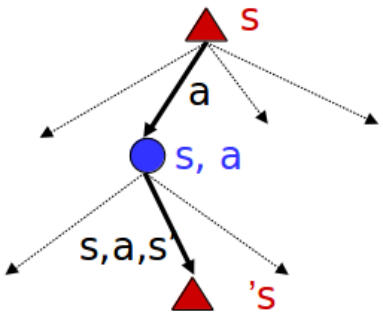


MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

POMDPs add:

POMDPs



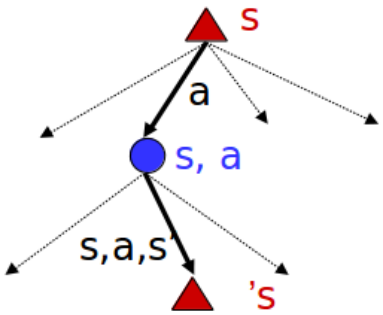
MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

POMDPs add:

- Observations O

POMDPs



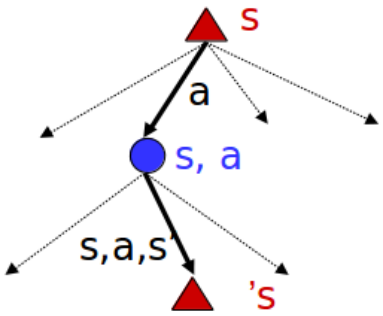
MDPs have:

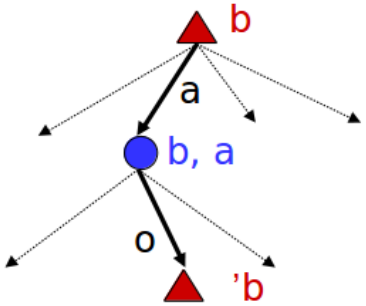
- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

POMDPs add:

- Observations O
- Observation function $P(o \mid s)$ (or $O(s, o)$)

POMDPs





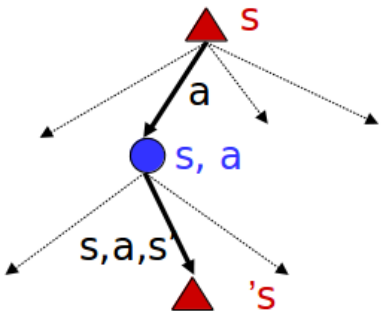
MDPs have:

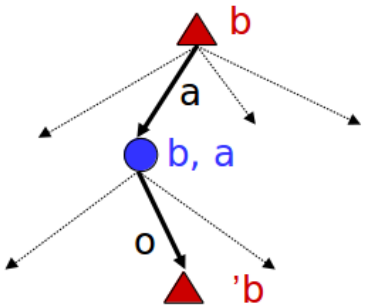
- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

POMDPs add:

- Observations O
- Observation function $P(o \mid s)$ (or $O(s, o)$)

POMDPs





MDPs have:

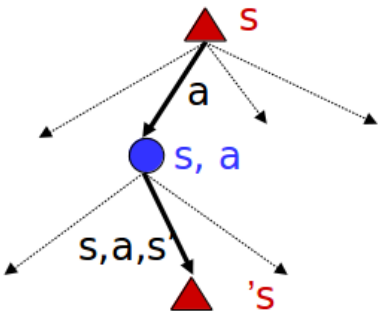
- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

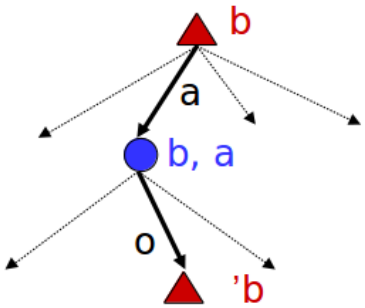
POMDPs add:

- Observations O
- Observation function $P(o \mid s)$ (or $O(s, o)$)

POMDPs are MDPs over belief states b (distributions over S)

POMDPs





MDPs have:

- States S
- Actions A
- Transition function P(s'| s,a) (or T(s,a,s'))
- Rewards R(s,a,s')

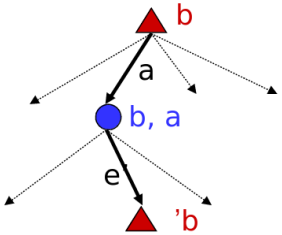
POMDPs add:

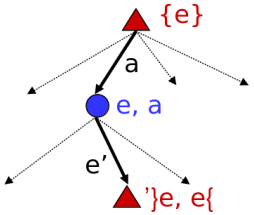
- Observations O
- Observation function $P(o \mid s)$ (or $O(s, o)$)

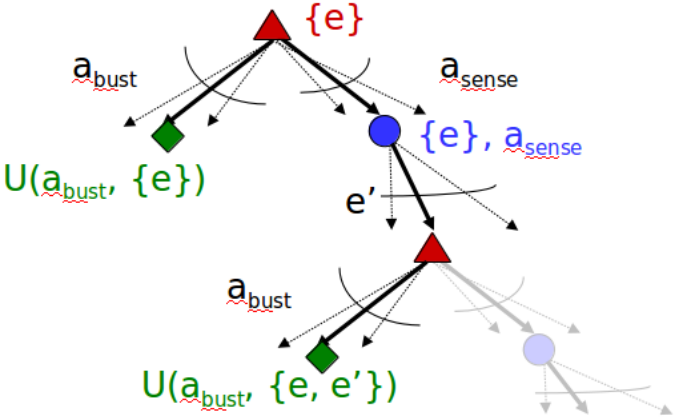
POMDPs are MDPs over belief states b (distributions over S)

This is like belief states in Hidden Markov Models!

Example: Ghostbusters

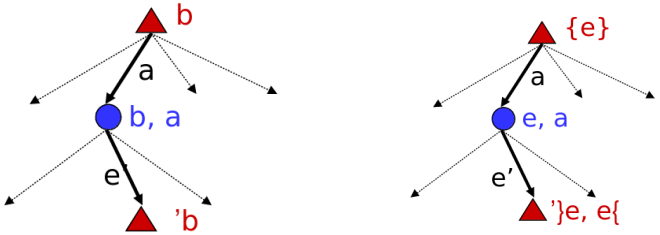


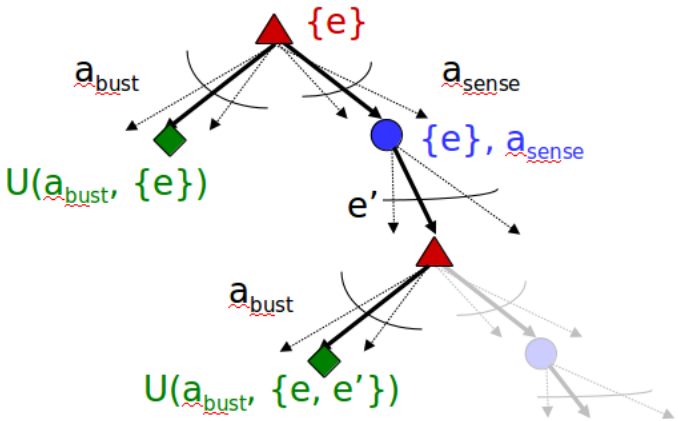




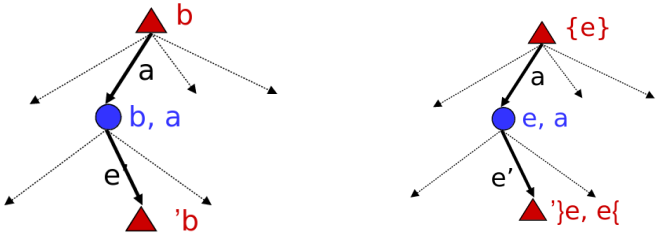
Example: Ghostbusters

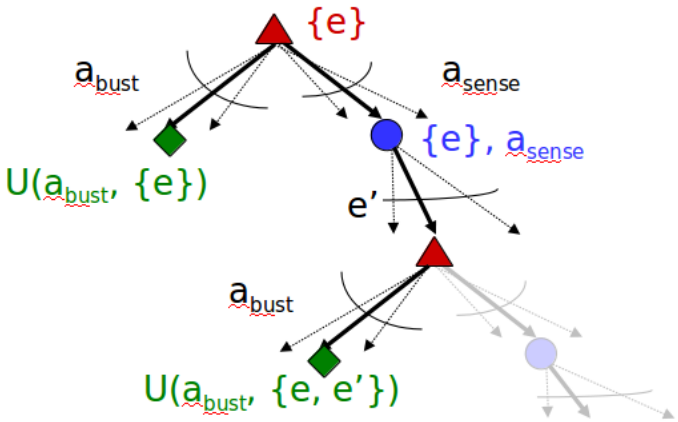
In (static) Ghostbusters:





Example: Ghostbusters

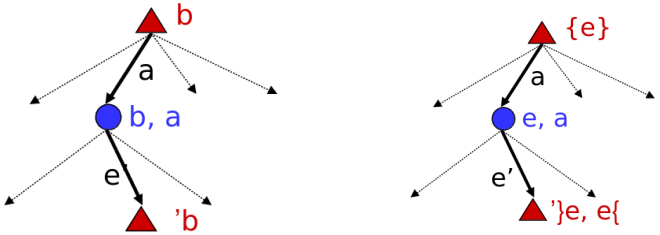


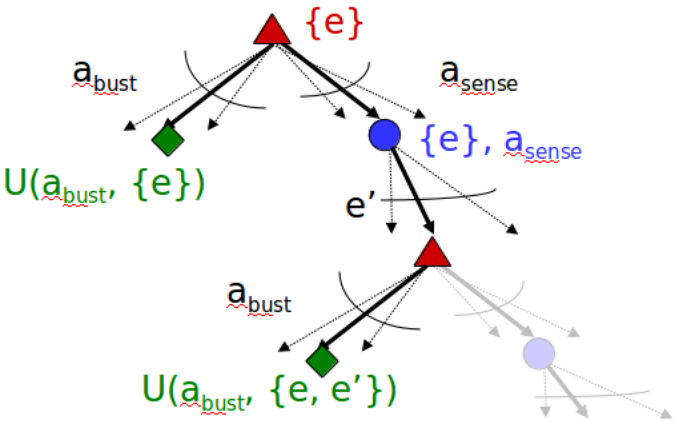


In (static) Ghostbusters:

- Belief state determined by evidence e.

Example: Ghostbusters

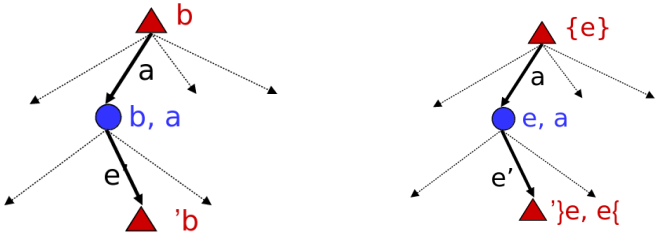




In (static) Ghostbusters:

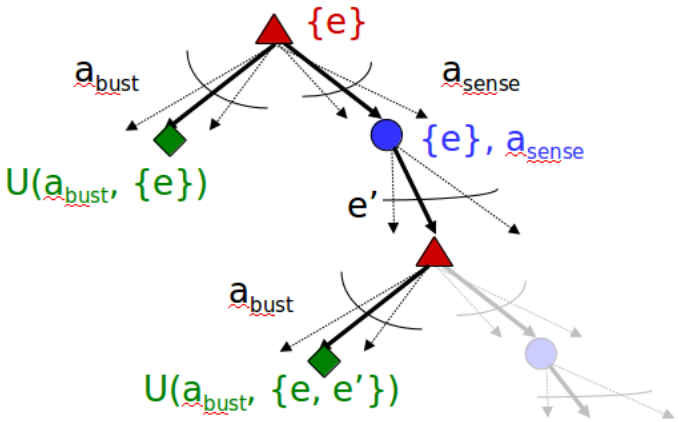
- Belief state determined by evidence e.
- Tree really over evidence sets.

Example: Ghostbusters

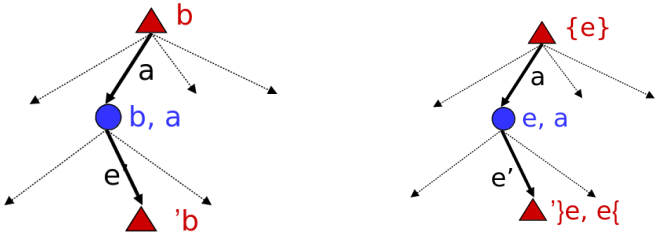


In (static) Ghostbusters:

- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

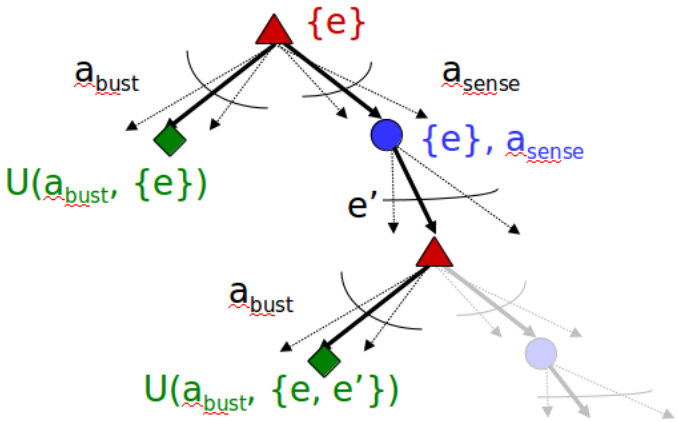


Example: Ghostbusters

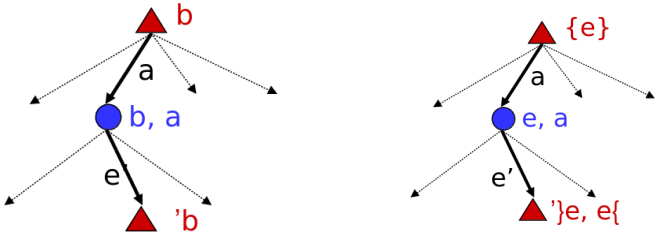


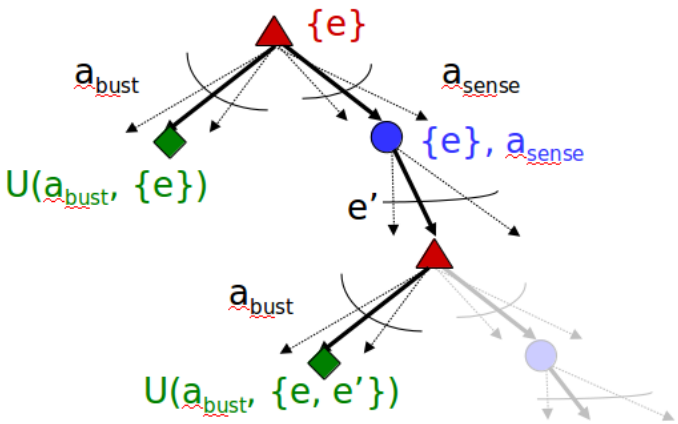
In (static) Ghostbusters:

- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence



Example: Ghostbusters



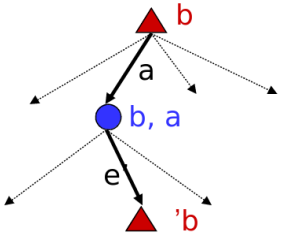


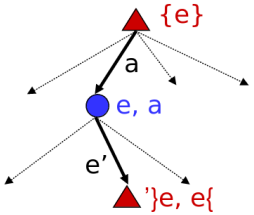
In (static) Ghostbusters:

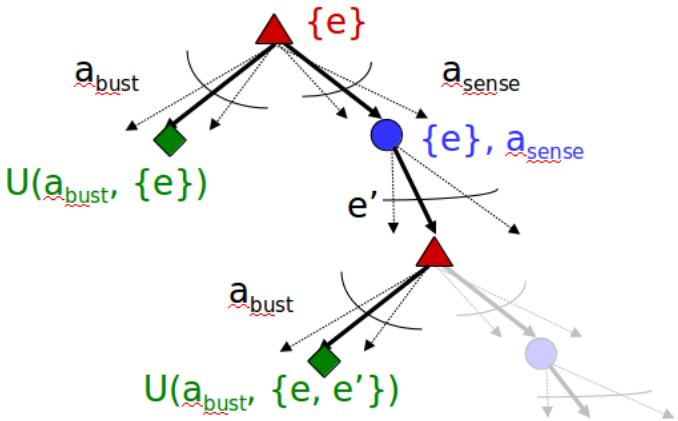
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

Solving POMDPs

Example: Ghostbusters







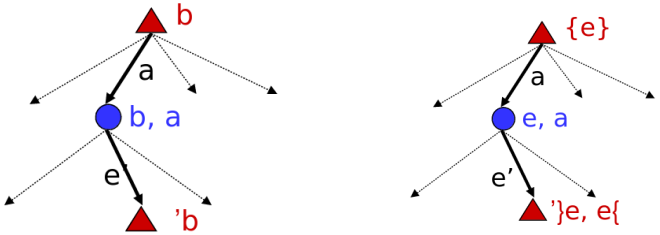
In (static) Ghostbusters:

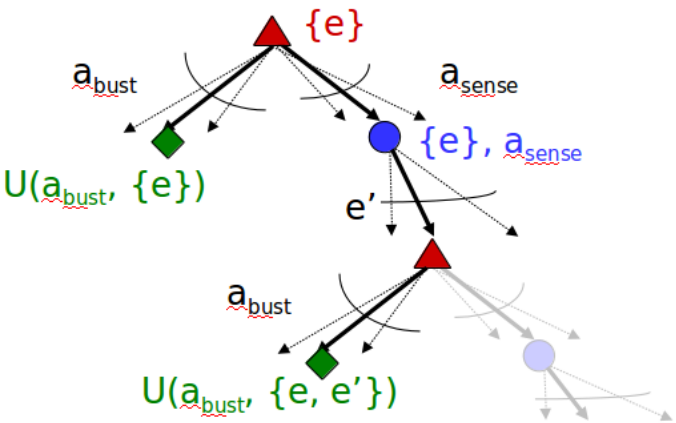
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

Solving POMDPs

- One way:

Example: Ghostbusters





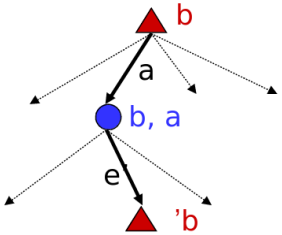
In (static) Ghostbusters:

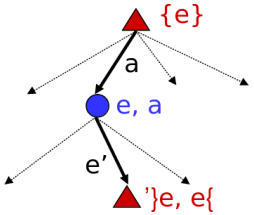
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

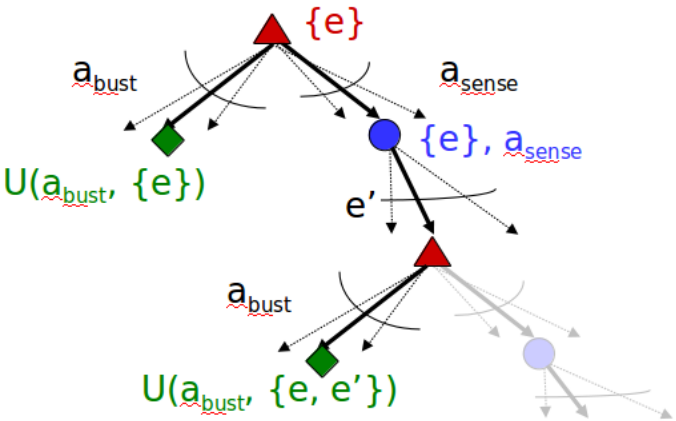
Solving POMDPs

- One way: use truncated expectimax to compute approximate value of actions

Example: Ghostbusters







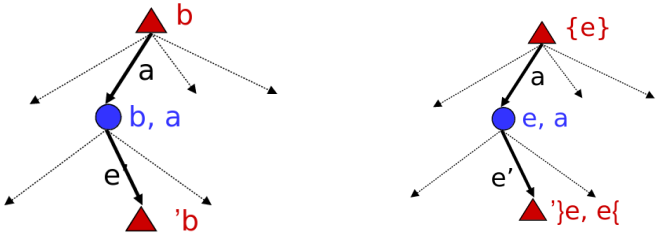
In (static) Ghostbusters:

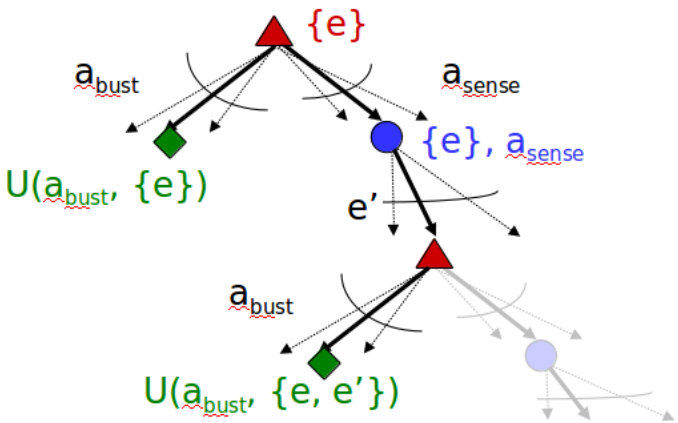
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

Solving POMDPs

- One way: use truncated expectimax to compute approximate value of actions
- What if you only considered busting or one sense followed by a bust?

Example: Ghostbusters





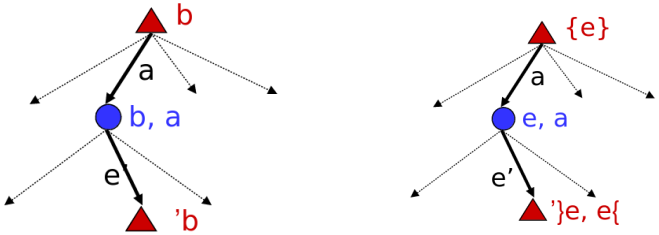
In (static) Ghostbusters:

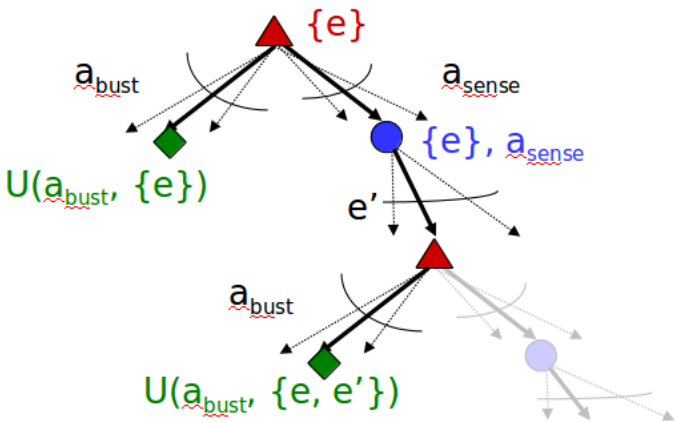
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

Solving POMDPs

- One way: use truncated expectimax to compute approximate value of actions
- What if you only considered busting or one sense followed by a bust?
- You get a VPI-based agent!

Example: Ghostbusters





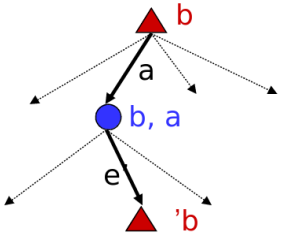
In (static) Ghostbusters:

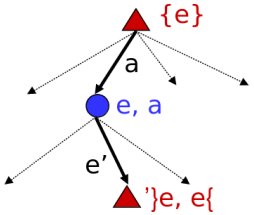
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

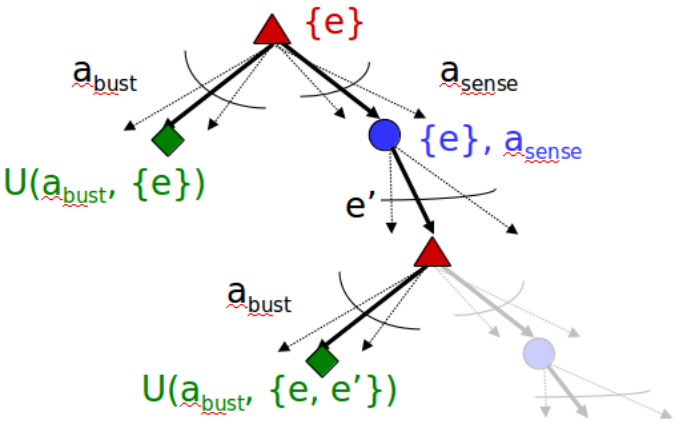
Solving POMDPs

- One way: use truncated expectimax to compute approximate value of actions
- What if you only considered busting or one sense followed by a bust?
- You get a VPI-based agent!

Example: Ghostbusters







In (static) Ghostbusters:

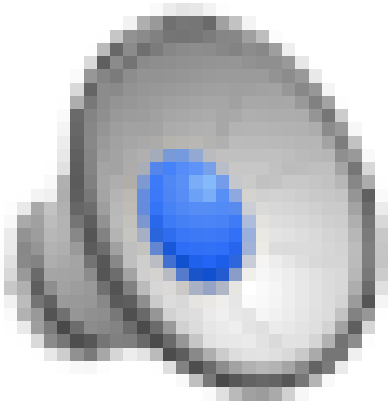
- Belief state determined by evidence e.
- Tree really over evidence sets.
- Probabilistic reasoning predicts new evidence given past evidence

Solving POMDPs

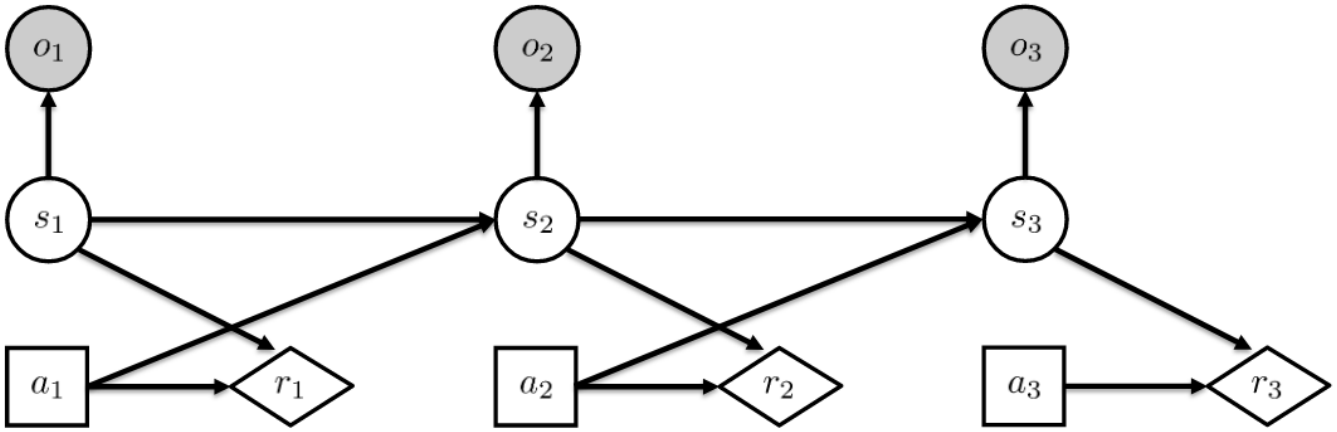
- One way: use truncated expectimax to compute approximate value of actions
- What if you only considered busting or one sense followed by a bust?
- You get a VPI-based agent!

Demo: Ghostbusters with VPI

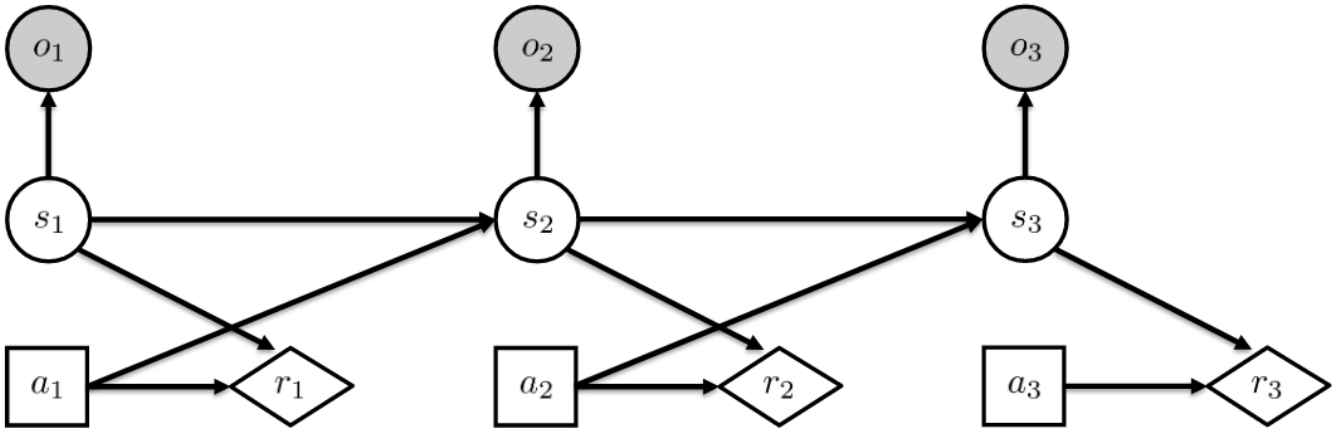
Video of Demo Ghostbusters with VPI



POMDPs as Decision Networks



POMDPs as Decision Networks

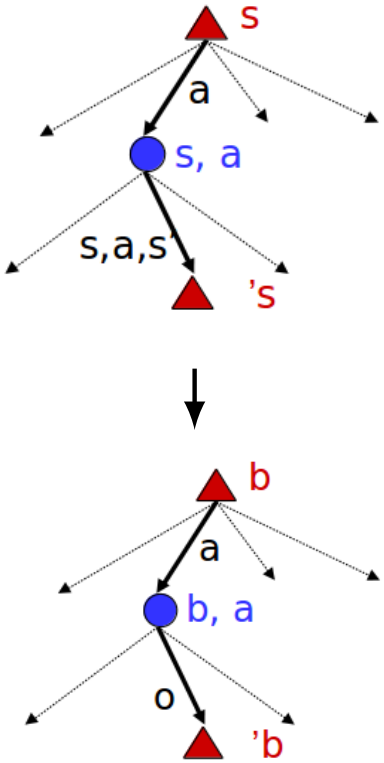


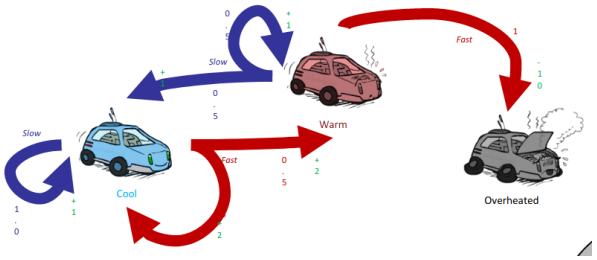
MDP:
States: S
Actions: A
Rewards: R(s,a,s')

POMCPs add:
Observations: O
Observation: $P(o|s)$

POMDPs More Generally*

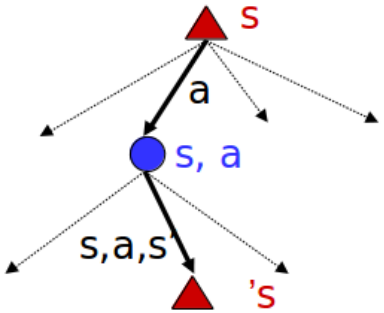
How can we solve POMDPs?

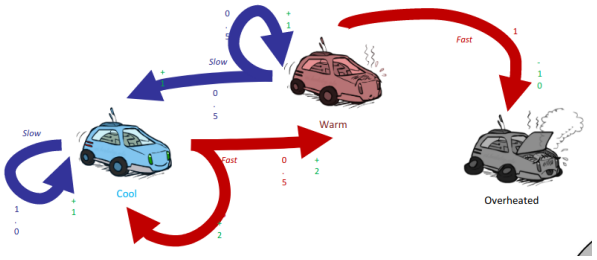


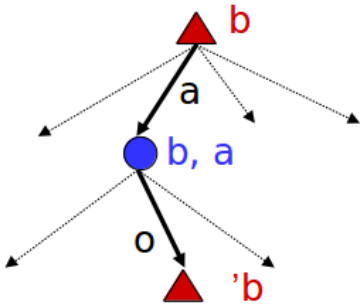


POMDPs More Generally*

How can we solve POMDPs?



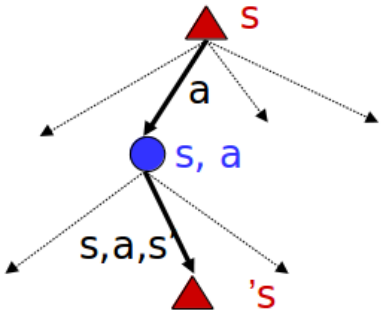


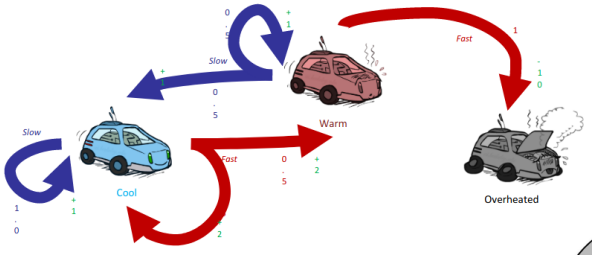


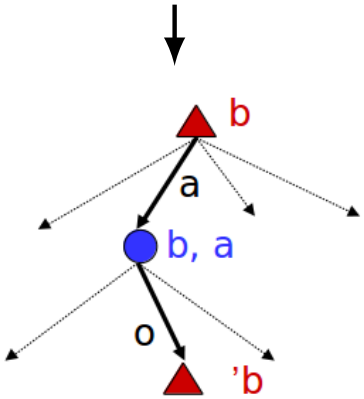
POMDP is MDP over *belief b*.

POMDPs More Generally*

How can we solve POMDPs?





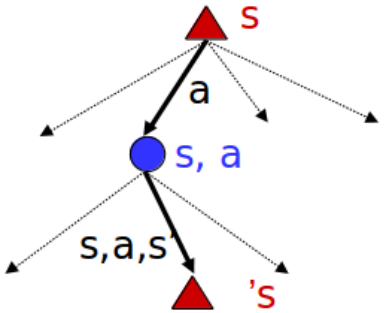


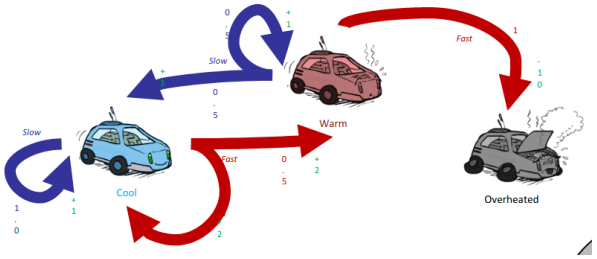
POMDP is MDP over *belief b*.

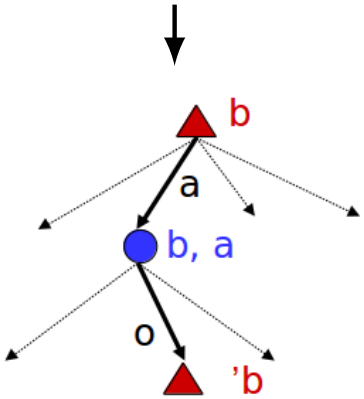
$s \in \{\text{cool, warm, overheated}\}$.

POMDPs More Generally*

How can we solve POMDPs?







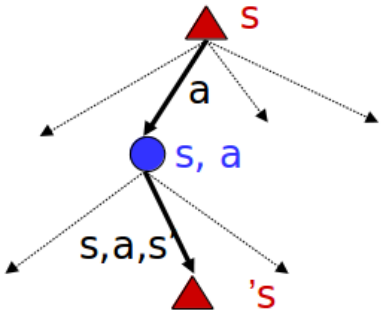
POMDP is MDP over *belief b*.

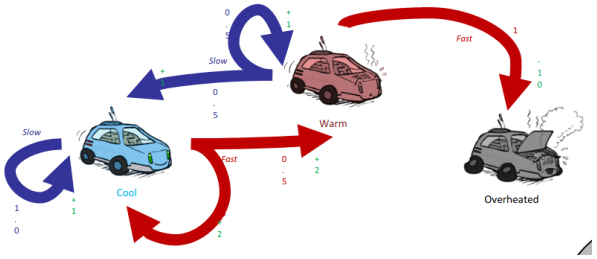
$s \in \{\text{cool, warm, overheated}\}$.

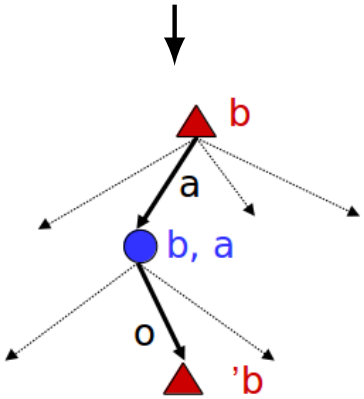
$b \in [0,1]^3 \leftarrow$ vector of three continuous numbers!

POMDPs More Generally*

How can we solve POMDPs?







POMDP is MDP over *belief b*.

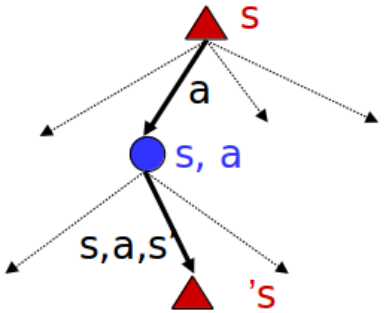
$s \in \{\text{cool, warm, overheated}\}$.

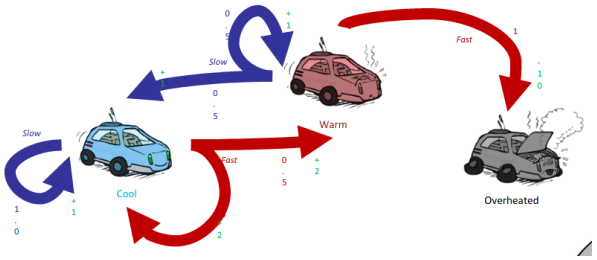
$b \in [0,1]^3 \leftarrow$ vector of three continuous numbers!

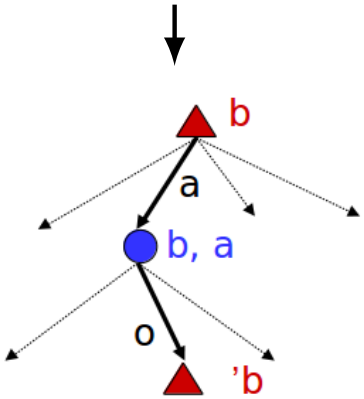
Use: Value iteration, policy iteration, etc. over b.

POMDPs More Generally*

How can we solve POMDPs?







POMDP is MDP over *belief b*.

$s \in \{\text{cool, warm, overheated}\}$.

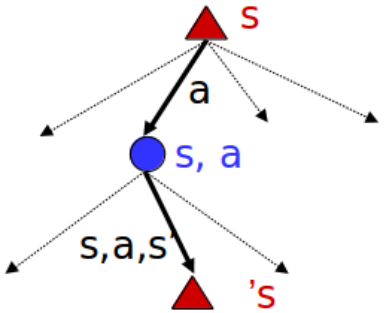
$b \in [0,1]^3 \leftarrow$ vector of three continuous numbers!

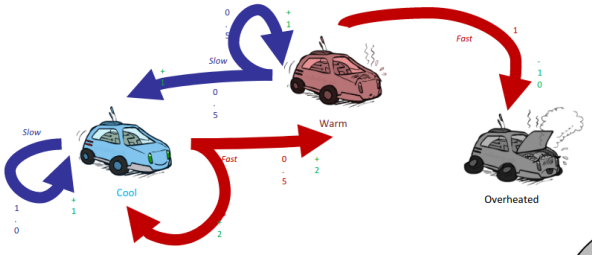
Use: Value iteration, policy iteration, etc. over b.

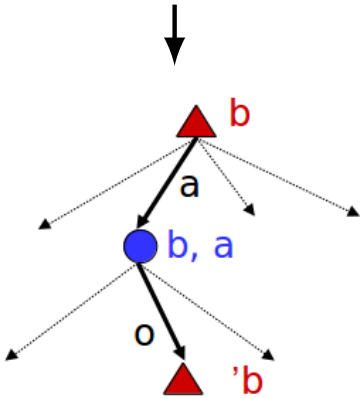
b id continuous (discretize or use functions.)

POMDPs More Generally*

How can we solve POMDPs?







POMDP is MDP over *belief b*.

$s \in \{\text{cool, warm, overheated}\}$.

$b \in [0,1]^3 \leftarrow$ vector of three continuous numbers!

Use: Value iteration, policy iteration, etc. over b.

b id continuous (discretize or use functions.)

b is very big (one number for each state.)

POMDPs More Generally*

POMDPs More Generally*

General solutions map belief functions to actions

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions)

POMDPs More Generally*

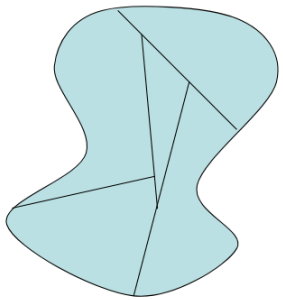
General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)

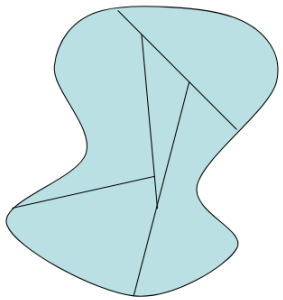


- Can build approximate policies using discretization methods

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)

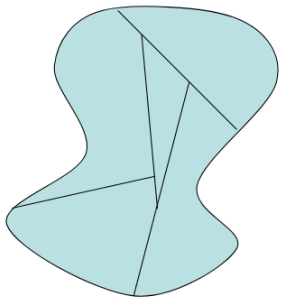


- Can build approximate policies using discretization methods
- Can factor belief functions in various ways

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)

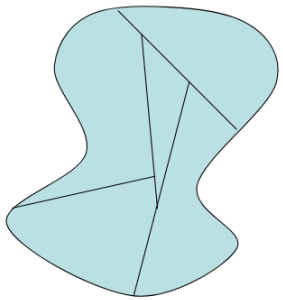


- Can build approximate policies using discretization methods
- Can factor belief functions in various ways

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)



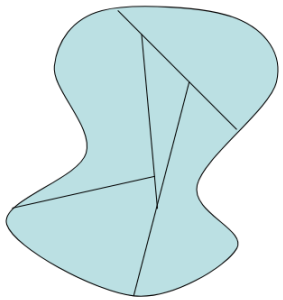
- Can build approximate policies using discretization methods
- Can factor belief functions in various ways

Overall, POMDPs are very (actually PSACE-) hard

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)



- Can build approximate policies using discretization methods
- Can factor belief functions in various ways

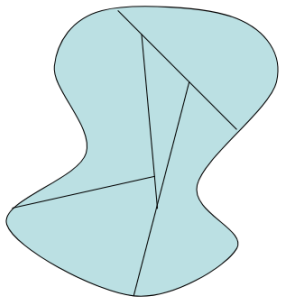
Overall, POMDPs are very (actually PSACE-) hard

Most real problems are POMDPs, but we can rarely solve them in general!

POMDPs More Generally*

General solutions map belief functions to actions

- Divide regions of belief space (set of belief functions) into policy regions (gets complex quickly)



- Can build approximate policies using discretization methods
- Can factor belief functions in various ways

Overall, POMDPs are very (actually PSACE-) hard

Most real problems are POMDPs, but we can rarely solve them in general! E.g. depth limited in ghostbusters.