CS 295A/395D: Artificial Intelligence

Decision Theory

Prof. Emma Tosch

28 March 2022

The University of Vermont

Agenda

Wrapping up reasoning under uncertain *state* today

Introduce taking actions in the presence of uncertainty.

Elementary decision theory

Elementary game theory (making decisions given a game) concepts

Logistics

- BB theory assignment out before next class
  - Update: bounties for real errors (will not be purposefully introducing errors)
  - Unlimited retries, soft deadline (recommended) before exam
- Plan to wrap up this unit by Friday.
- Next programming assignment out by the end of the week
- Exam next week (Wednesday or Friday, to give you time to study)

Recap: KT45^n syntax & Semantics

$\top \mid \bot \mid p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \rightarrow \varphi_2$
$\mid K_i\varphi \mid E_G\varphi \mid C_G\varphi \mid D_G\varphi$

Given Model structure: $\mathcal{M} = (W, \{R\}_{\{i \in A\}}, L)$.

$w \Vdash \top$, $w \nVdash \bot$, $w \Vdash a$ iff $a \in L(w)$, $w \Vdash \neg\varphi$ iff $w \nVdash \varphi$

$w \Vdash \varphi \wedge \psi$ iff $w \Vdash \varphi$ and $w \Vdash \psi$

$w \Vdash \varphi \vee \psi$ iff at least one of $w \Vdash \varphi$ or $w \Vdash \psi$

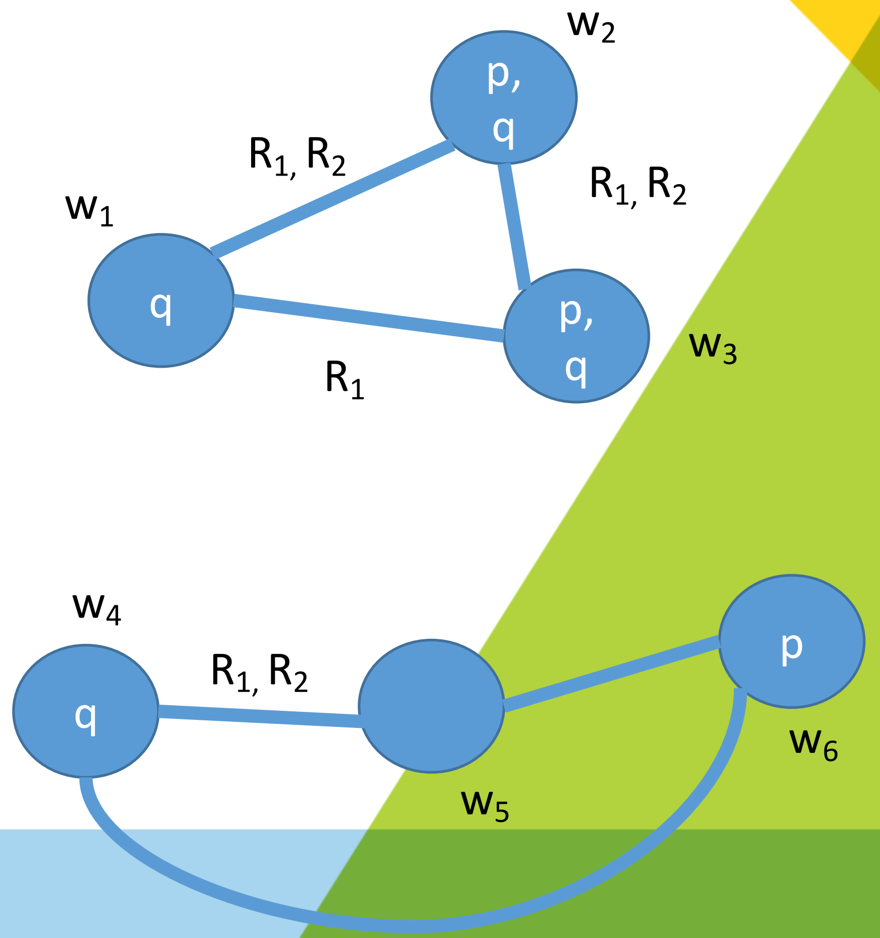
$w \Vdash \varphi \rightarrow \psi$ iff $w \Vdash \varphi$ whenever $w \Vdash \psi$

$w \Vdash K_i \varphi$ iff $\forall$ $w' \in W$ $(R_i(w, w') \rightarrow w' \Vdash \psi)$

$w \Vdash E_G\varphi$ iff $\forall$ $i \in G$ $(w \Vdash Ki\, \psi)$

$w \Vdash C_G\varphi$ iff $\forall$ $k \geq 1$ $(w \Vdash E_G^k\, \psi)$

$w \Vdash D_G\varphi$ iff $\forall$ $w' \in W$ $(\forall$ $i \in G$ $(R_i\,(w, w') \rightarrow w \Vdash \psi)$



Recap: Evaluating formulas in KT45^n

Given Model structure: $\mathcal{M} = (W, \{R\}_{\{i \in A\}}, L)$.

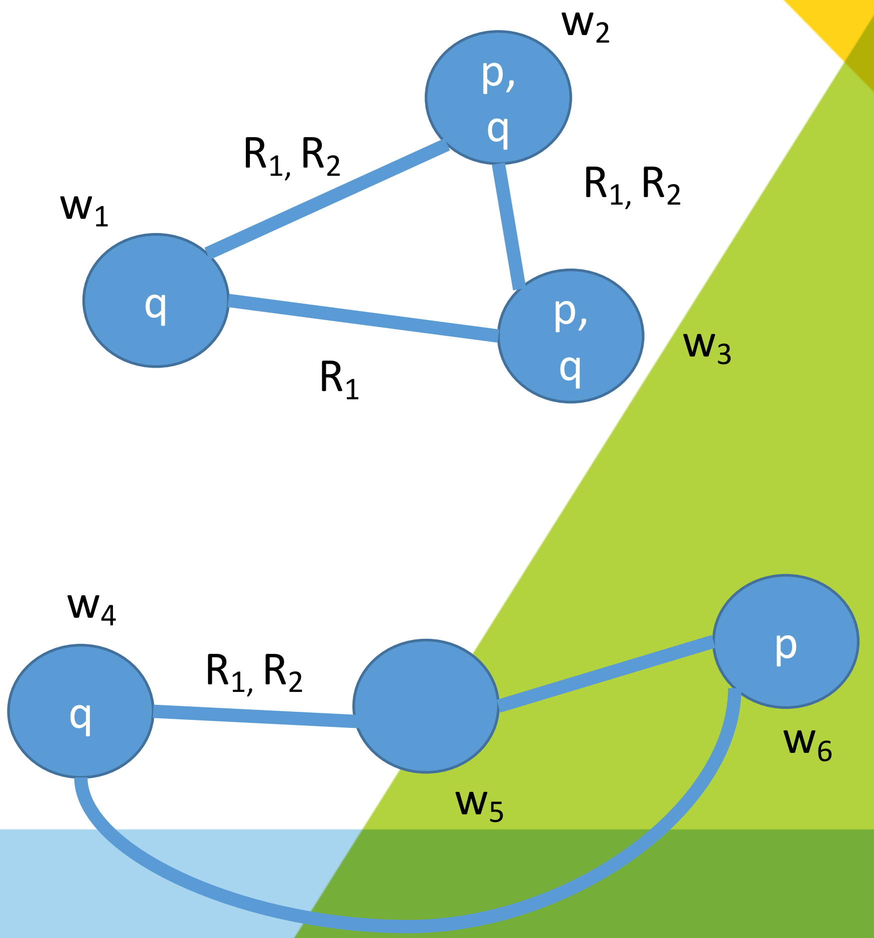
Does $w_3 \Vdash K_1 p$?

- Recall: $w \Vdash K_i\, \varphi$ iff $\forall$ w' $\in$ W (R_i(w, w') $\rightarrow$ w' $\Vdash \varphi$)
- For each w in R_1(w_3, w), does w $\Vdash$ p?
  - First world to check: w_3 (by axiom T)
  - $w_3 \nVdash p$

Does $w_3 \Vdash K_2 p$?

- $w_3 \Vdash$ p?
- $w_2 \Vdash$ p?

Does $w_3 \Vdash K_3 p$?



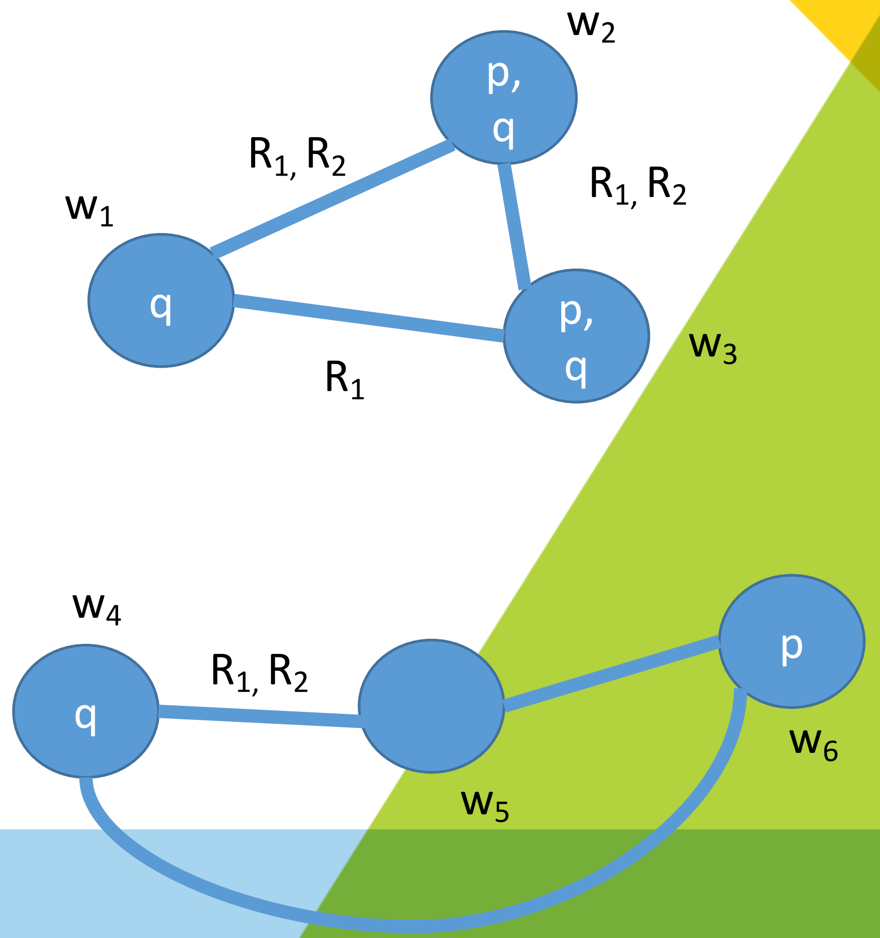
Recap: Evaluating formulas in KT45^n

Given Model structure: $\mathcal{M} = (W, \{R\}_{\{i \in A\}}, L)$.

Does $w_3 \Vdash K_1 K_2 p$?

- Recall: $w \Vdash K_i\, \varphi$ iff $\forall\, w' \in W\ (R_i(w, w') \rightarrow w' \Vdash \varphi)$
- For each w in $R_1(w_3, w)$, does $w_3 \Vdash K_2 p$?
  - $w_3 \Vdash K_2$ p?
  - $w_1 \Vdash K_2$ p?

Agent 1 doesn't know if agent 2 knows p.



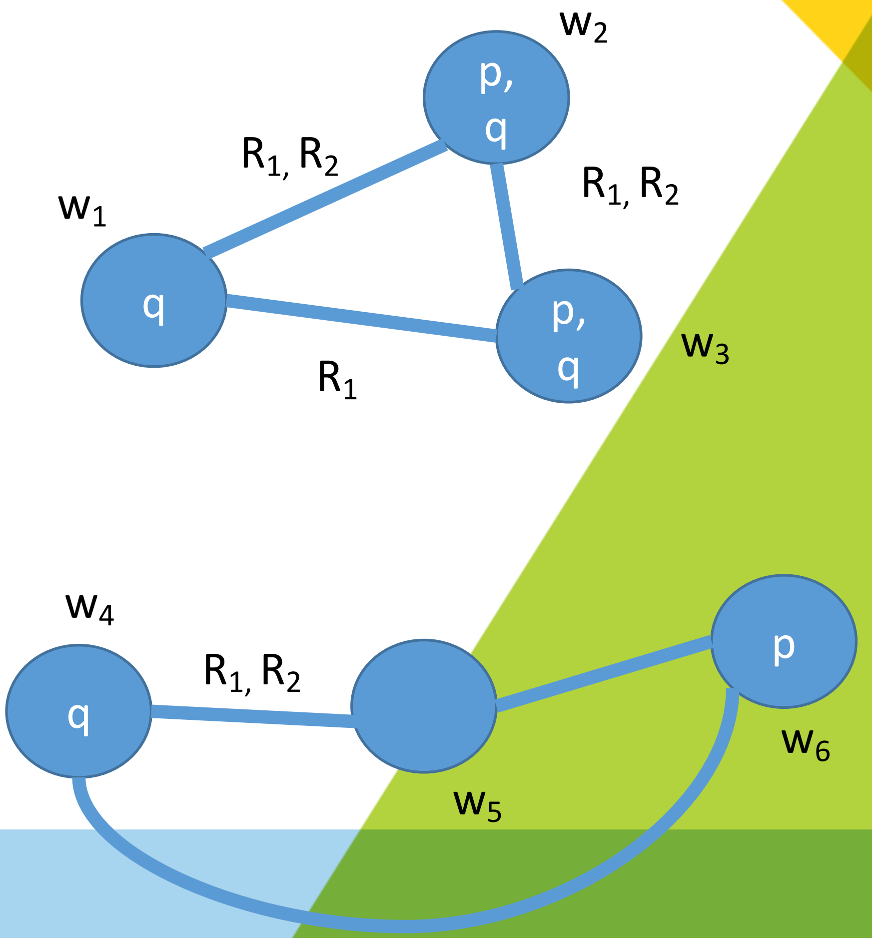
Recap: Evaluating formulas in KT45^n

Given Model structure: $\mathcal{M} = (W, \{R\}_{\{i \in \mathrm{A}\}}, L)$.

Does $w_3 \Vdash K_2 K_1 p$?

- Recall: $w \Vdash K_i\, \varphi$ iff $\forall\, w' \in W\ (R_i(w, w') \rightarrow w' \Vdash \varphi)$
- For each w in $R_2(w_3, w)$, does $w_3 \Vdash K_1 p$?
  - $w_3 \Vdash K_1$ p?

Agent 2 doesn't know if agent 1 knows p.



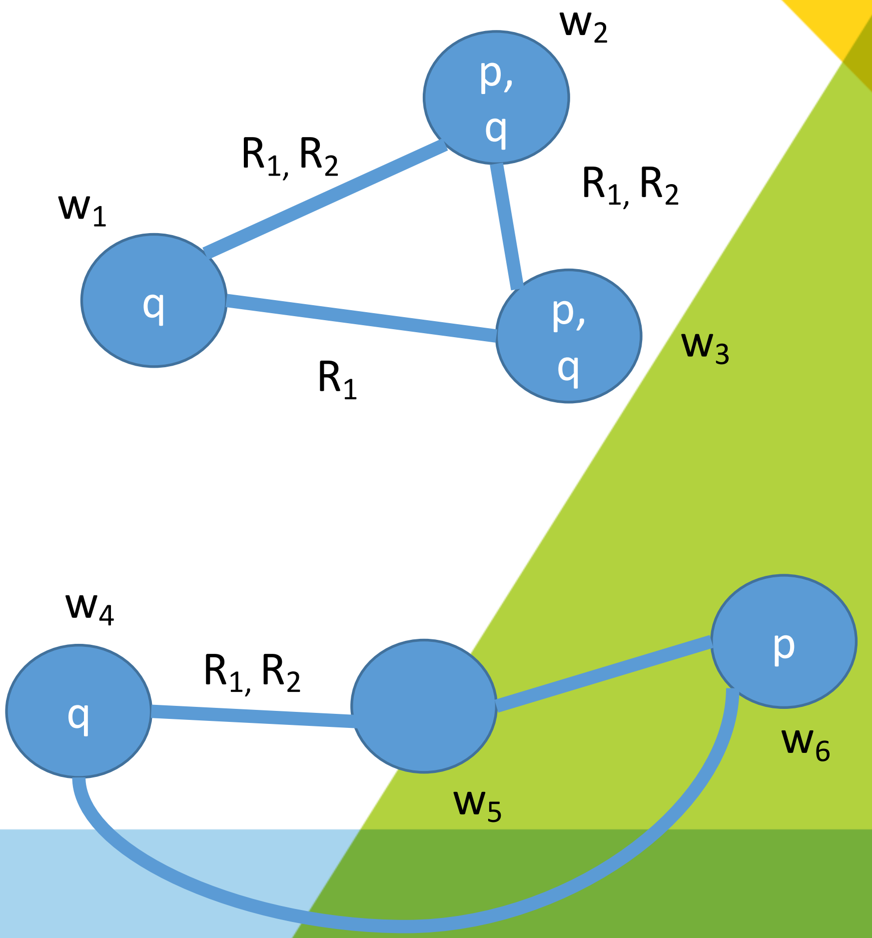
Recap: Evaluating formulas in KT45^n

Given Model structure: $\mathcal{M} = (W, \{R\}_{\{i \in A\}}, L)$.

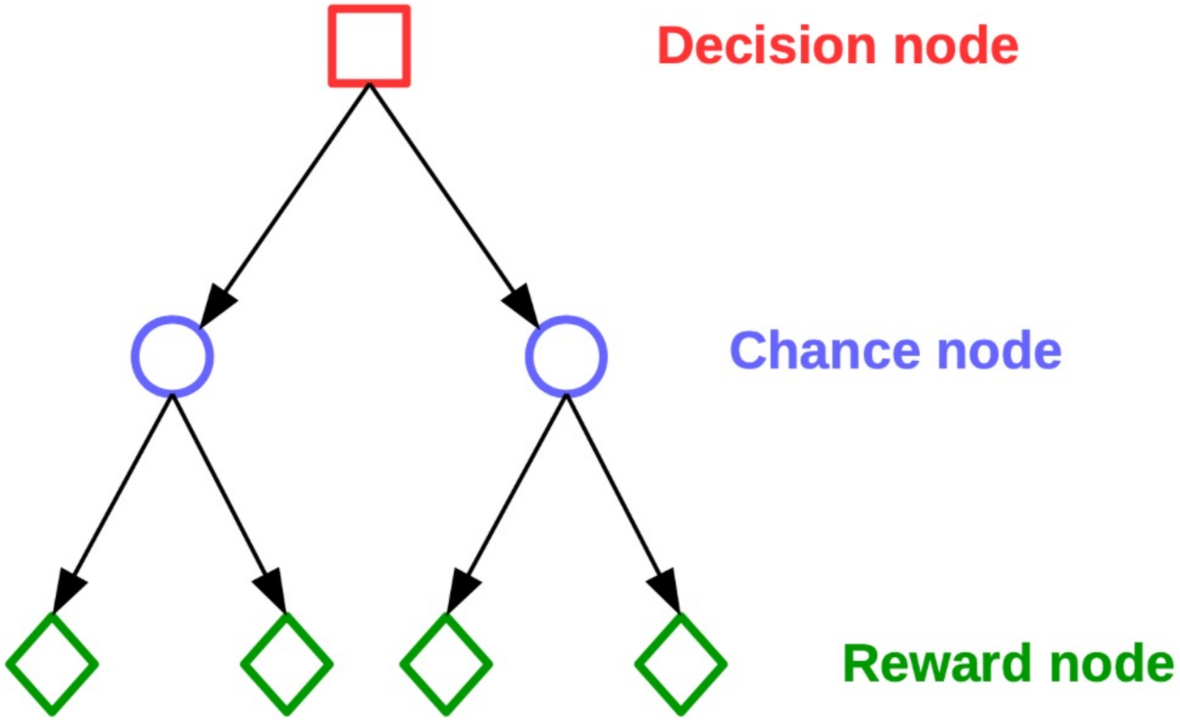
Does $w_3 \Vdash K_3 K_2 p$?

- Recall: $w \Vdash K_i\, \varphi$ iff $\forall\, w' \in W\ (R_i(w, w') \rightarrow w' \Vdash \varphi)$
- For each w in $R_3(w_3, w)$, does $w_3 \Vdash K_2 p$?
  - $w_3 \Vdash K_2\, p$?

Agent 3 can't access worlds 2 or 1, so it can only reason over world 3, where it knows that agent 2 knows *p*.



Basic Decision Theory

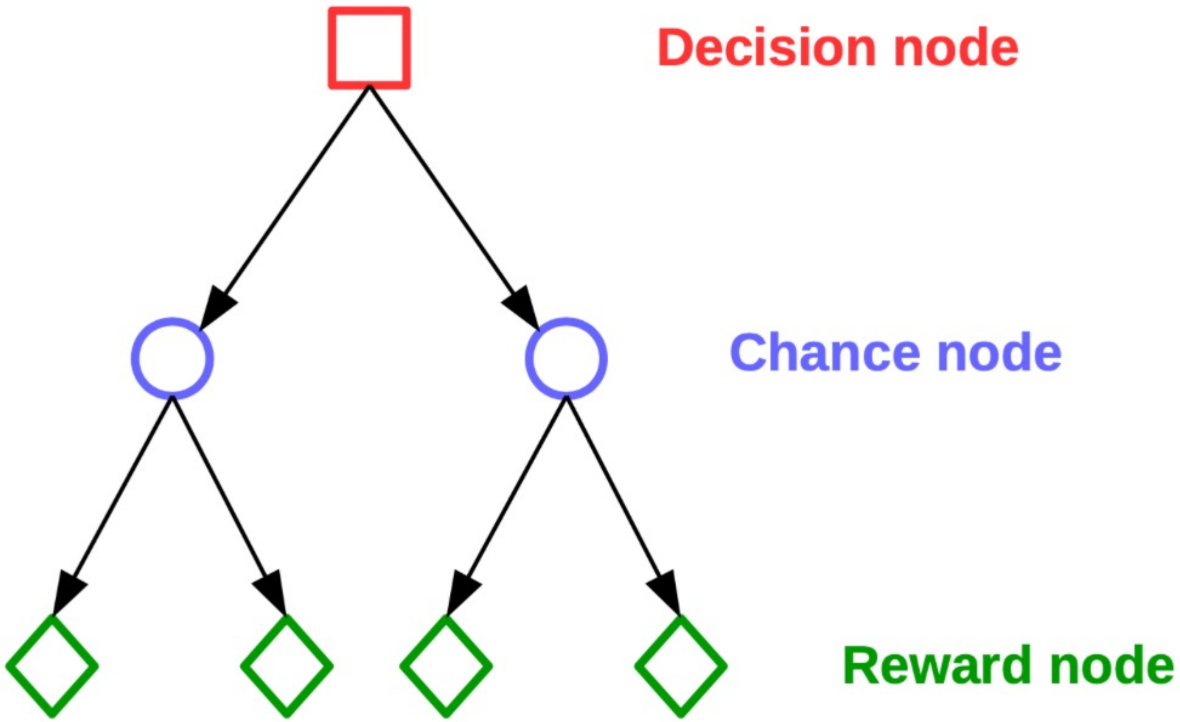


Note: "decision tree" also refers to a classification algorithm in machine learning and is completely different from the type of decision tree we will talk about here.

We can express taking actions in a world with uncertainty via *decision trees*

Decisions trees are temporally-ordered nodes where each level corresponds to alternating:

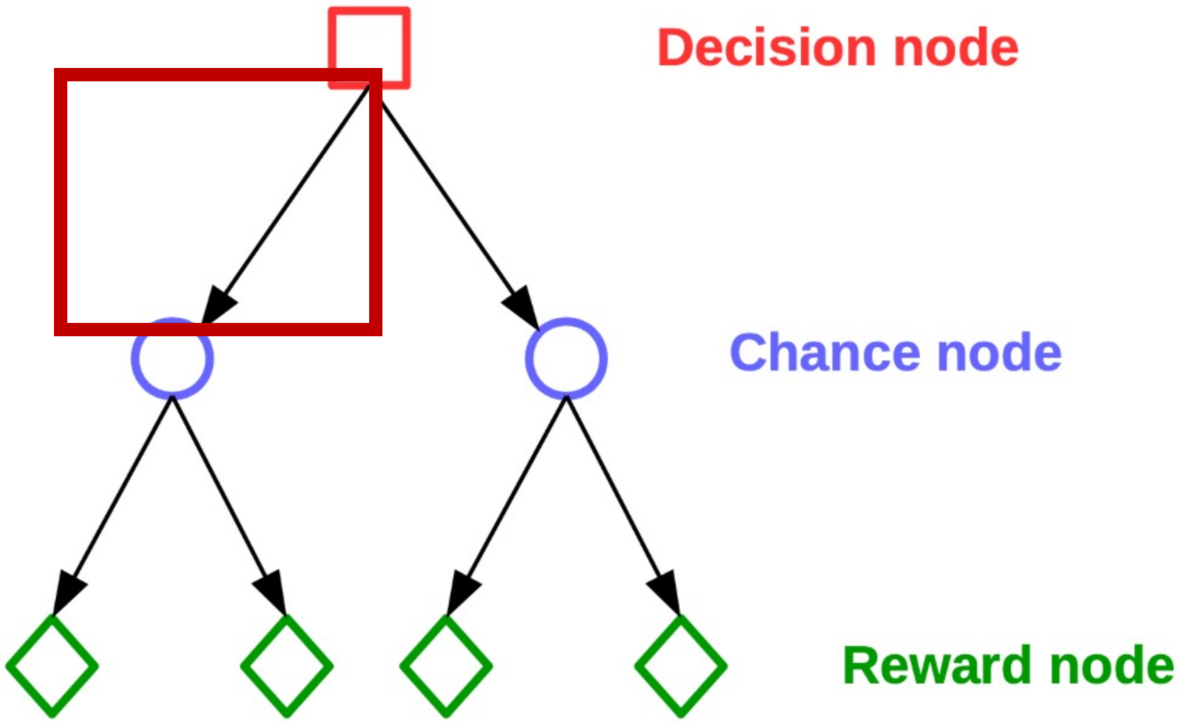
- **Decision nodes –** state of the system; outgoing edges represent different actions
- **Chance nodes –** probability distributions over outcomes; outgoing edges represent reachable states with some probability
- **Reward nodes** – utility obtained from following the path



Note: "decision tree" also refers to a classification algorithm in machine learning and is completely different from the type of decision tree we will talk about here.

Goal: Maximize expected utility

$$EU[\, a \mid e_1, e_2, \dots \,] = \sum_{s'} P(\, S_{t+1} = s' \mid a, e_1, e_2, \dots)\, U(s')$$



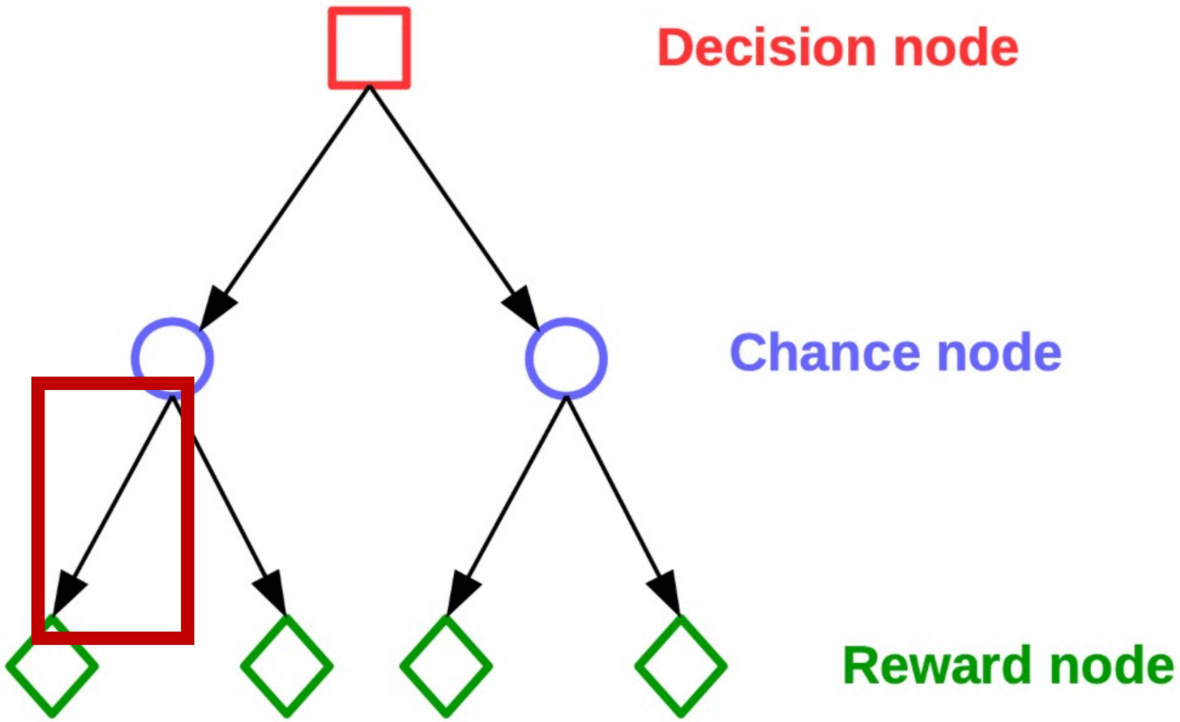
Note: "decision tree" also refers to a classification algorithm in machine learning and is completely different from the type of decision tree we will talk about here.

Goal: Maximize expected utility

$$EU[\, a \mid e_1, e_2, \ldots \,] = \sum_{s'} P(\, S_{t+1} = s' \mid a, e_1, e_2, \ldots) \, U(s')$$

Best action is the action a that maximizes

$$EU[\, a \mid e_1, e_2, \ldots \,]$$



Note: "decision tree" also refers to a classification algorithm in machine learning and is completely different from the type of decision tree we will talk about here.

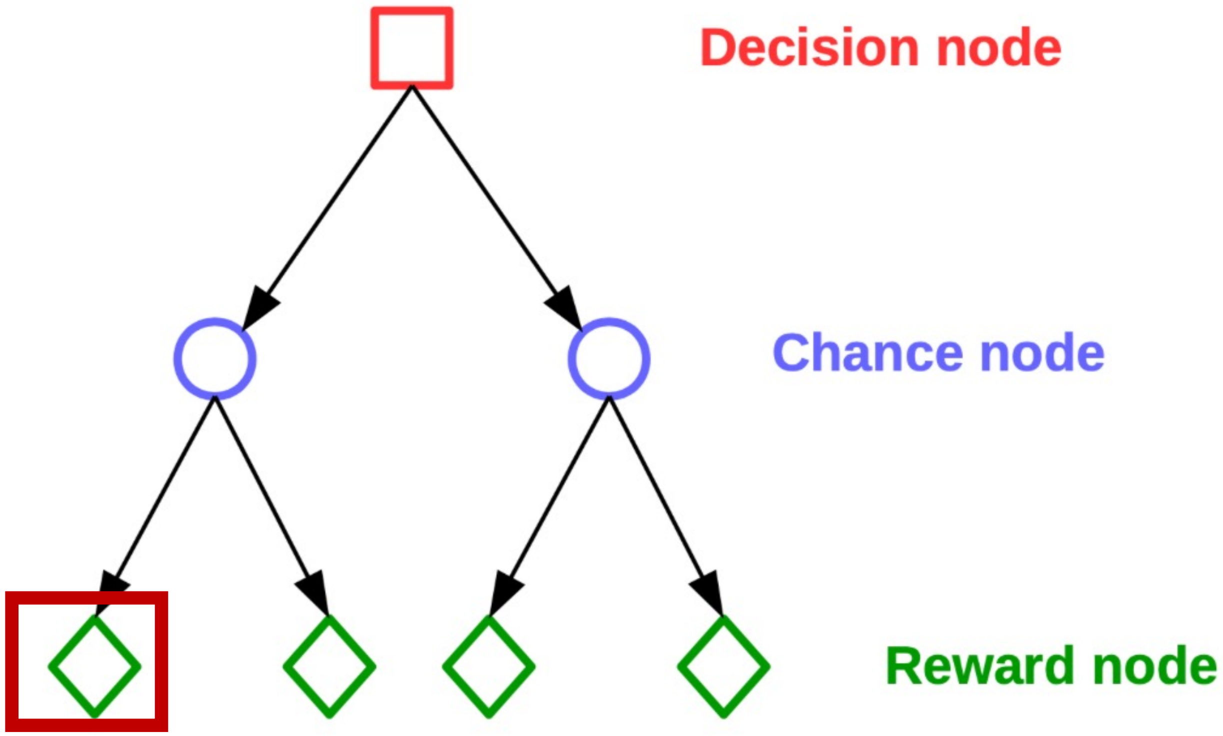
Goal: Maximize expected utility

$$EU[\,a \;\; e_1 \;\; e_2, \ldots\,] = \sum_{s'} P(\,S_{t+1} = s' \mid a \;\; e_1 \;\; e_2, \ldots\,)\,U(s')$$

Best action is the action a that maximizes

$$EU[\,a \mid e_1, e_2, \ldots\,]$$

Goal: Maximize expected utility



$$EU[\, a \mid e_1, e_2, \ldots \,] = \sum_{s'} P(\, S_{t+1} = s' \mid a, e_1, e_2, \ldots) \, U(s')$$

Best action is the action a that maximizes $EU[\, a \mid e_1, e_2, \ldots \,]$

Sum of the utility of actions taken.

Note: "decision tree" also refers to a classification algorithm in machine learning and is completely different from the type of decision tree we will talk about here.

Example: Buying a car

Problem setup:

- Choice between 2 cars (C_1 and C_2), which can each be good (+) or bad (-) quality
- Two tests that cost $$: (T_1 : $50, T_2: $20)
- C_1 costs $500 below m.v., but if it's bad quality, must be repaired for $700
- C_2 costs $250 below m.v., but if it's bad quality, must be repaired for $150
- Must buy exactly one car and can perform at most one test before buying.

$P(C_1 = +) = 0.7$ $P(C_2 = +) = 0.8$

Example: Buying a car

Problem setup:

- **Choice between 2 cars (C_1 and C_2), which can each be good (+) or bad (-) quality**
- Two tests for each car that cost $$: (T_1 : $50, T_2: $20)
- C_1 costs $500 below m.v., but if it's bad quality, must be repaired for $700
- C_2 costs $250 below m.v., but if it's bad quality, must be repaired for $150
- Must buy exactly one car and can perform at most one test before buying.

Example: Buying a car

$P(C_1 = +) = 0.7$ $P(C_2 = +) = 0.8$

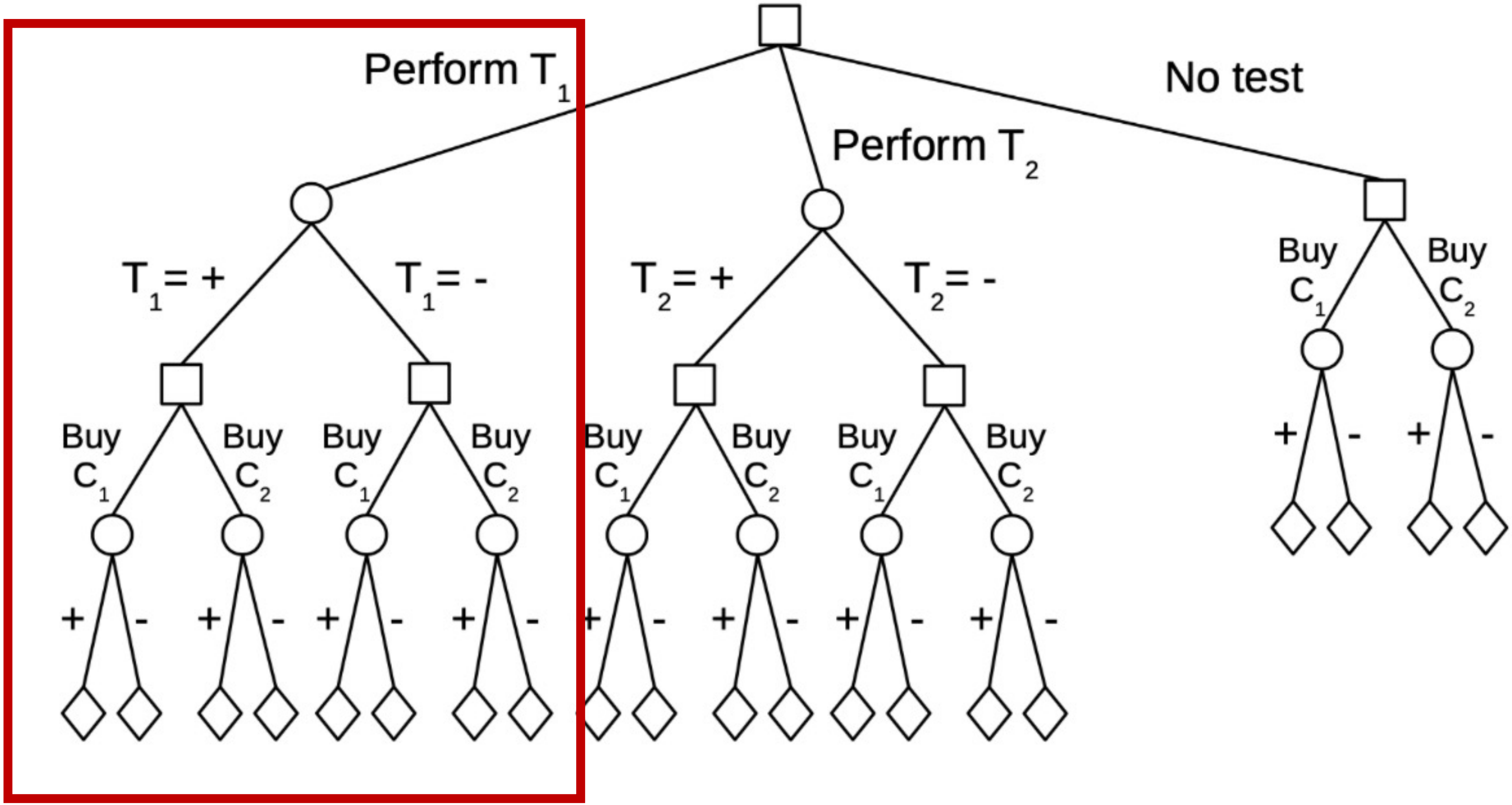
$P(T_1 = + \mid C_1 = +) = 0.8$ $P(T_1 = - \mid C_1 = -) = 0.65$

$P(T_2 = + \mid C_2 = +) = 0.75$ $P(T_2 = - \mid C_2 = -) = 0.7$

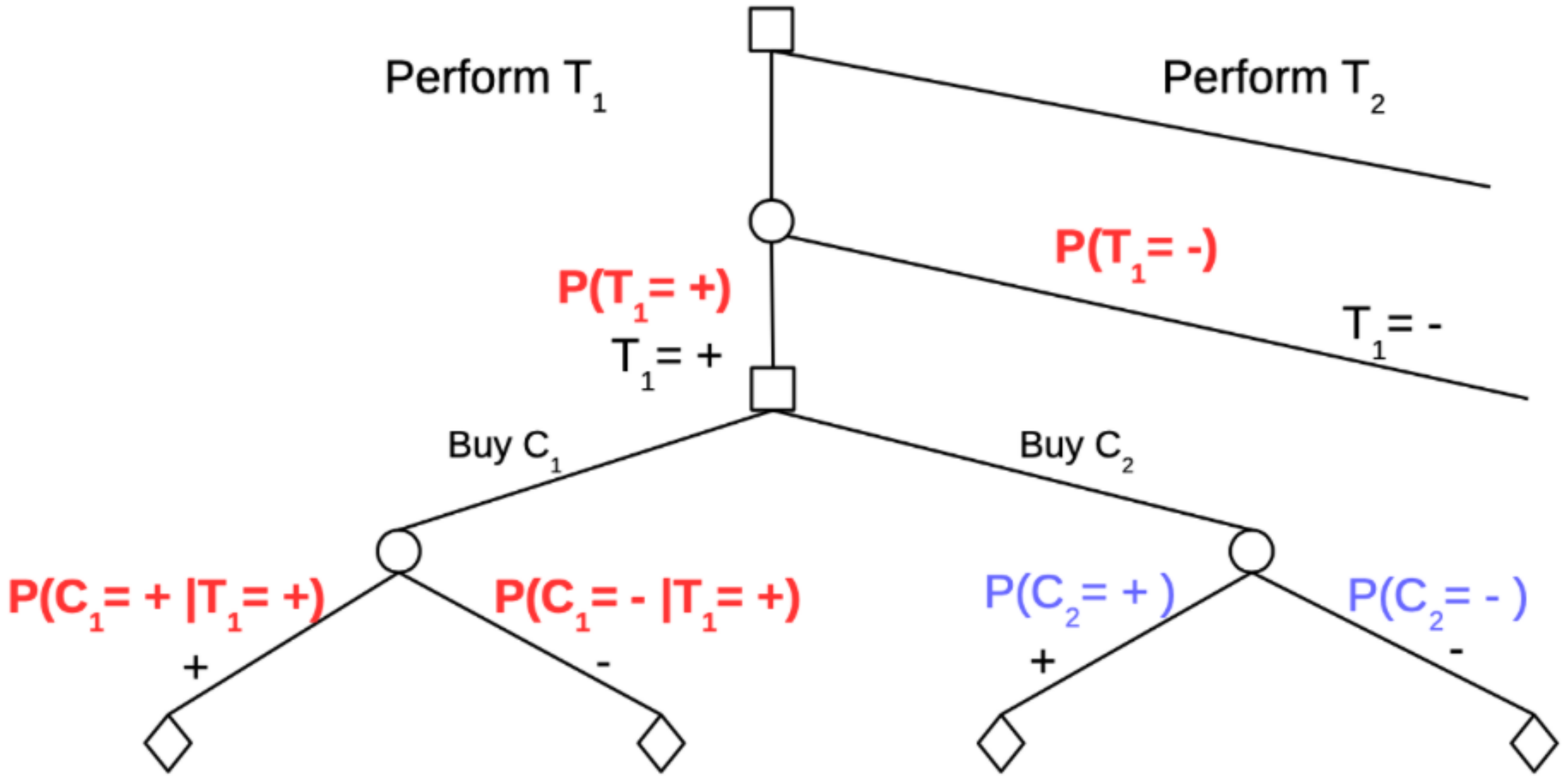
Problem setup:

- Choice between 2 cars (C_1 and C_2), which can each be good (+) or bad (-) quality
- **One test for each car that cost \$\$: (T_1 : \$50, T_2: \$20)**
- C_1 costs \$500 below m.v., but if it's bad quality, must be repaired for \$700
- C_2 costs \$250 below m.v., but if it's bad quality, must be repaired for \$150
- Must buy exactly one car and can perform at most one test before buying.

What are the expected costs of C_1 and C_2 assuming you run no tests? (Board 1)



Perform T1
Perform T2
No test
T1= +
T1= -
T2= +
T2= -
Buy C1
Buy C2
+
-



What is the expected cost of buying C_1 and C_2 assuming you run T_1 *on C_1?*
Set up decision between T_1 and T_2 (Board 2)