CS 480

Introduction to Artificial Intelligence

November 8, 2022

Announcements / Reminders

- Follow Week 11 TO DO List
- Written Assignment #03 due on ~~Sunday (11/06/22)~~ Thursday (11/10) at 11:00 PM CST
- Programming Assignment #02 due on Sunday (11/20/22) at 11:00 PM CST
- Quiz #03 posted due on Sunday (11/13/22) at 11:00 PM CST
- Grading TA assignment:

  https://docs.google.com/spreadsheets/d/1ExS0bKnGt_fdf4LHa3YS1qRA7-Iq4xqXVjfSAPMaGVk/edit?usp=sharing

- UPDATED Final Exam date:
  - December 1st, 2022 (last week of classes!)
    - Ignore the date provided by the Registrar

Plan for Today

- Bayesian/Belief Networks
- Inference in Bayesian Networks
- Decision Networks

Bayesian (Belief) Network

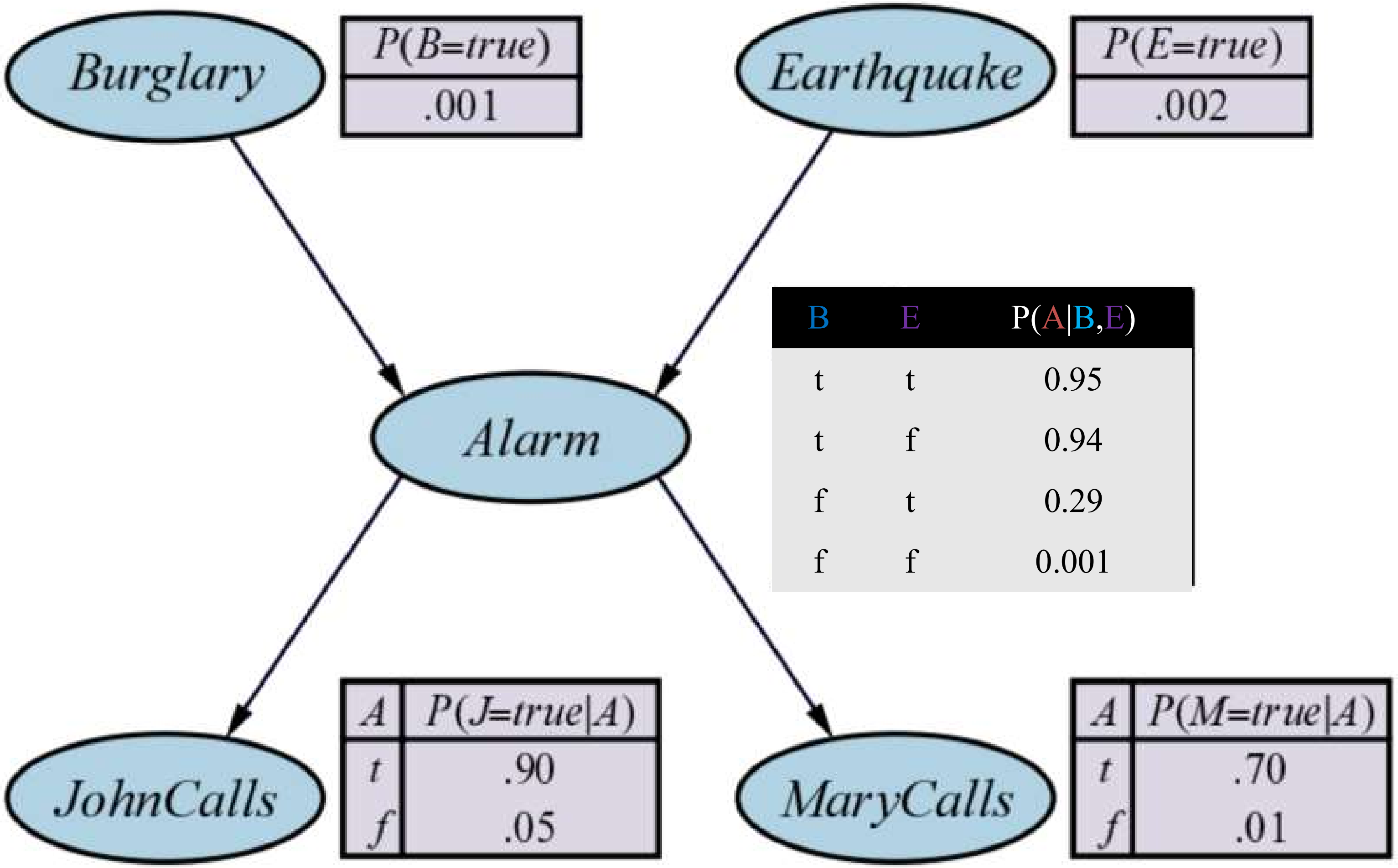
A Bayesian belief network describes the joint probability distribution for a set of variables.

A Bayesian network is an acyclic, directed graph (DAG), where the nodes are random variables (propositions). There is an edge (arc) from each elements of $\mathrm{parents}(X_i)$ into X_i. Associated with the Bayesian network is a set of conditional probability distributions - the conditional probability of each variable given its parents (which includes the prior probabilities of those variables with no parents).

Consists of:

- a graph (DAG) with nodes corresponding to random variables
- a domain for each random variable
- a set of conditional probability distributions $P(X_i \mid \mathrm{parents}(X_i))$

Add Conditional Probability Tables



P(B=true)
.001

P(E=true)
.002

B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

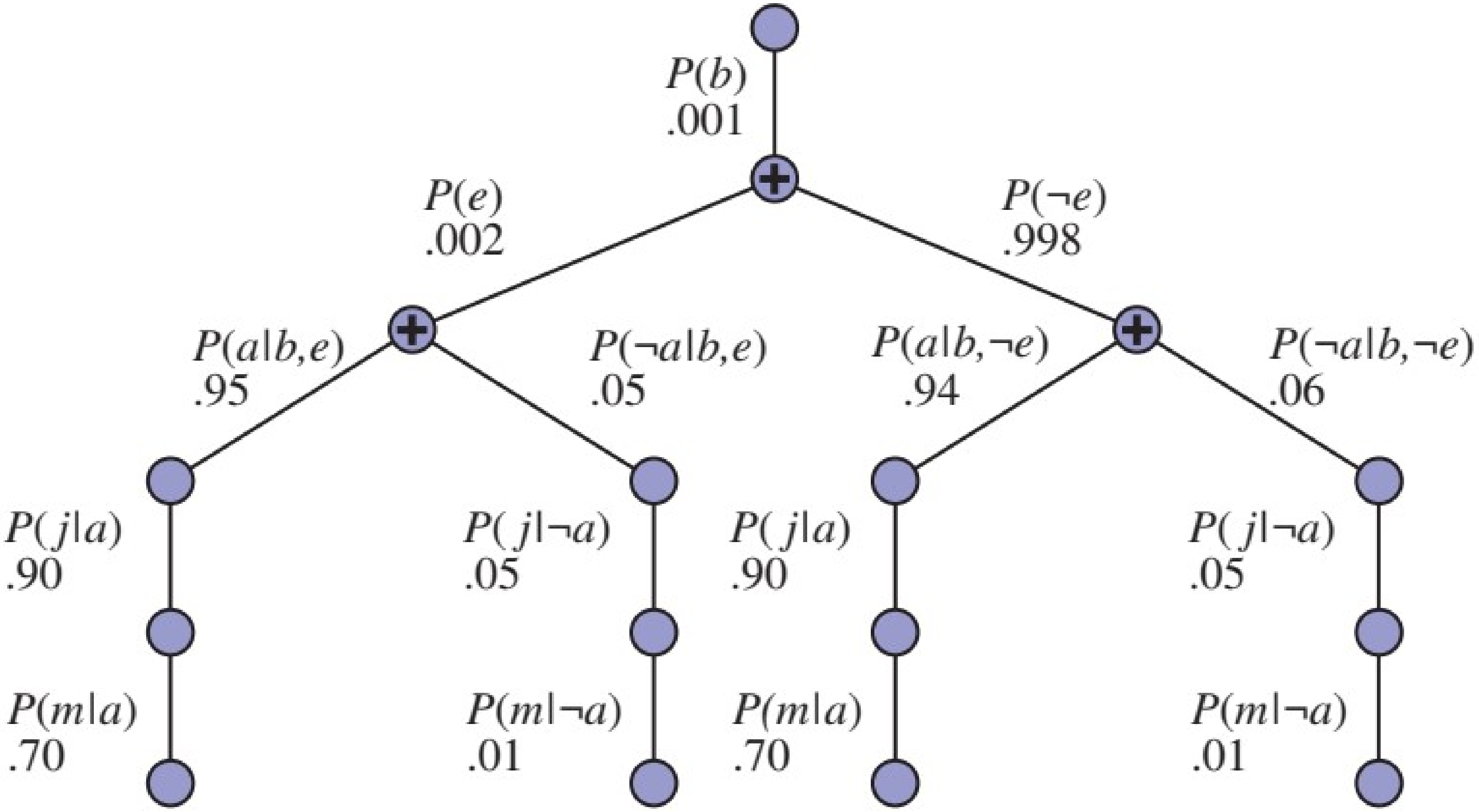
A	P(J=true|A)
t	.90
f	.05

A	P(M=true|A)
t	.70
f	.01

Inference by Enumeration: Example

Query:

$$\boldsymbol{P}(Burglary \mid JohnCalls\ =\ true \land MaryCalls\ =\ true)$$



General Inference Procedure

Given:

- **a query involving a single variable X (in our example: Cavity),**
- **a list of evidence variables E (in our example: just Toothache),**
- **a list of observed values e for E,**
- **a list of remaining unobserved variables Y (in our example: just Catch),**

where X, E, and Y together are a COMPLETE set of variables for the domain, the probability P(X | E) can be evaluated as:

$$\boldsymbol{P}(X \mid \boldsymbol{e}) \ = \ \alpha * \boldsymbol{P}(X, \boldsymbol{e}) \ = \ \alpha * \sum_{\boldsymbol{y}} \boldsymbol{P}(X, \boldsymbol{e}, \boldsymbol{y})$$

where ys are all possible values for Ys, α - normalization constant.

P(X, e, y) is a subset of probabilities from the joint distribution

Inference

Query:

$$P(Burglary \mid JohnCalls = true \wedge MaryCalls = true)$$

Given:

- **a query involving a single variable** X
- **a list of evidence variables** $\mathbf{K}$**,**
- **a list of observed values** $\mathbf{k}$ **for** $\mathbf{K}$**,**
- **a list of remaining unobserved variables** $\mathbf{Y}$

the probability $\mathrm{P}(\mathrm{X} \mid \mathbf{K})$ **can be evaluated as:**

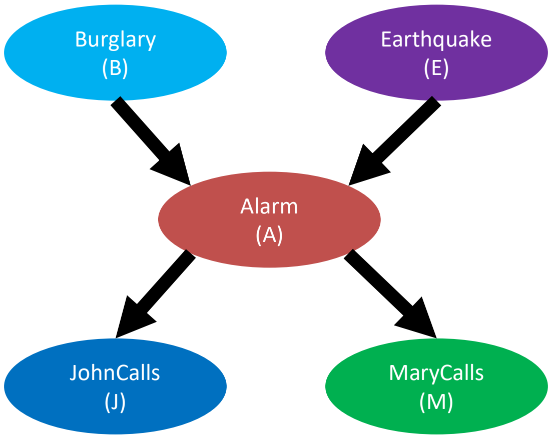
$$\boldsymbol{P}(X \mid \boldsymbol{k}) = \alpha * \boldsymbol{P}(X, \boldsymbol{k})$$

$$= \alpha * \sum_{\boldsymbol{y}} \boldsymbol{P}(X, \boldsymbol{k}, \boldsymbol{y})$$

where y**s are all possible values for** Y**s,** α **- normalization constant.**

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query:

$$\boldsymbol{P}(Burglary \mid JohnCalls = true \wedge MaryCalls = true)$$

Given:

- **a query involving a single variable** X**:** $Burglary$
- **a list of evidence variables** $\mathbf{K}$**:** $JohnCalls$, $MaryCalls$
- **a list of observed values** k **for** $\mathbf{K}$**:** $johnCalls$, $maryCalls$
- **a list of remaining unobserved variables** $\mathbf{Y}$**:** $Earthquake$, $Alarm$

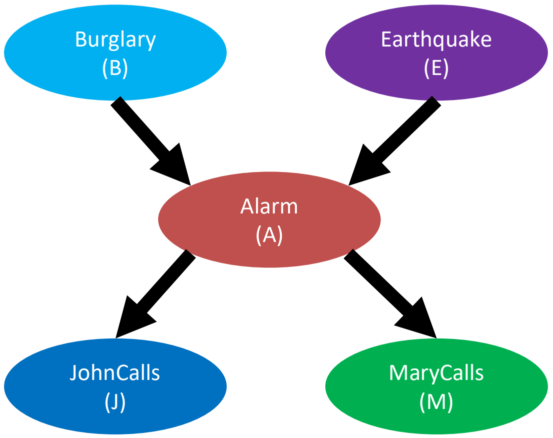
the probability $\mathrm{P}(\mathrm{X} \mid \mathbf{K})$ **can be evaluated as:**

$$\boldsymbol{P}(X \mid \boldsymbol{k}) = \alpha * \sum_{\boldsymbol{y}} \boldsymbol{P}(X, \boldsymbol{k}, \boldsymbol{y})$$

where y**s are all possible values for** Y**s,** α **- normalization constant.**

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query:

$$P(Burglary \mid JohnCalls = true \wedge MaryCalls = true)$$

Given:

- a query involving a single variable X: B
- a list of evidence variables K: J, M
- a list of observed values k for K: j, m
- a list of remaining unobserved variables Y: E, A

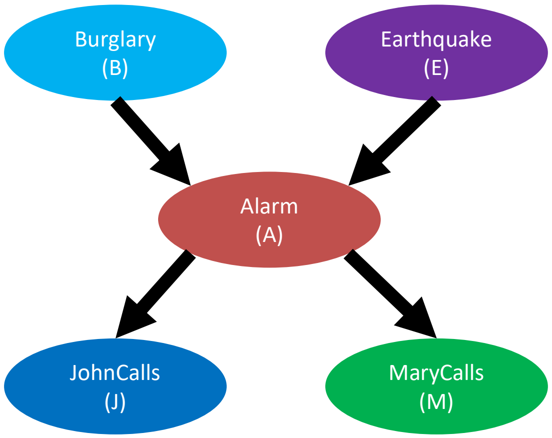
the probability P(X | K) can be evaluated as:

$$\boldsymbol{P}(X \mid \boldsymbol{k}) = \alpha * \sum_{\boldsymbol{y}} \boldsymbol{P}(X, \boldsymbol{k}, \boldsymbol{y})$$

where ys are all possible values for Ys, α - normalization constant.

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query:

$$\boldsymbol{P}(Burglary \mid JohnCalls = true \wedge MaryCalls = true)$$

Given:

- **a query involving a single variable B**
- **a list of evidence variables K: J, M**
- **a list of observed values k for K: j, m**
- **a list of remaining unobserved variables Y: E, A**

the probability P(B | J, M) can be evaluated as:

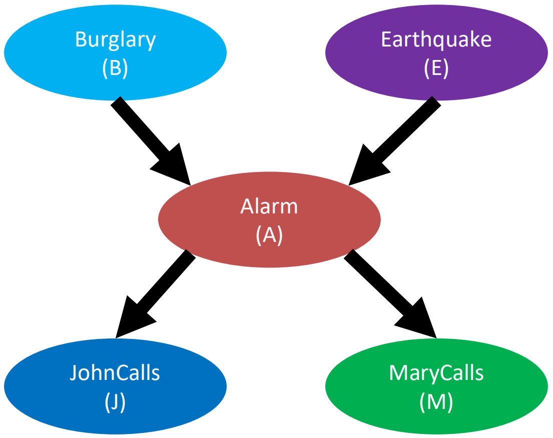
$$\boldsymbol{P}(B \mid j, m)$$

$$= \alpha * \sum_{e} \sum_{a} \boldsymbol{P}(B, j, m, e, a)$$

where ys are all possible values for Ys, α - normalization constant.

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let's change it a bit for simplicity):

$\boldsymbol{P}(Burglary = true \mid JohnCalls = true \land MaryCalls = true)$

Given:

- a query involving a single variable B
- a list of evidence variables $\mathbf{K}$: J, M
- a list of observed values $\mathbf{k}$ for $\mathbf{K}$: j, m
- a list of remaining unobserved variables $\mathbf{Y}$: E, A

the query can be evaluated as:

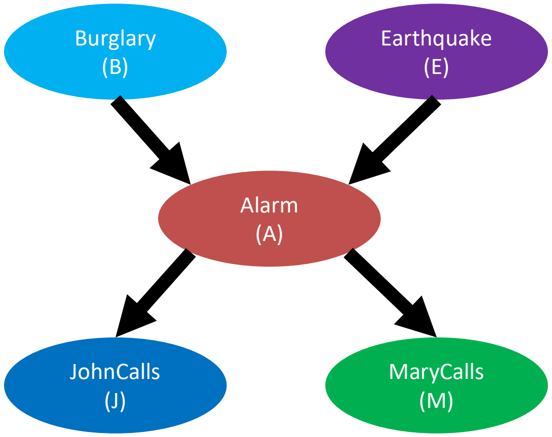
$$P(b \mid j, m) = \alpha * \sum_{e} \sum_{a} P(b, j, m, e, a)$$

By Chain rule:

$$P(b, j, m, e, a) = P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let's change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

Given:

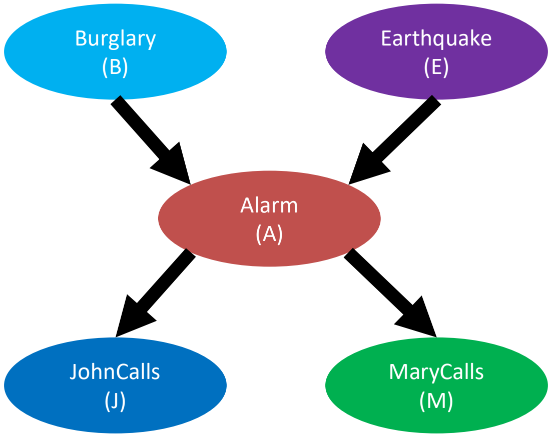
- **a query involving a single variable B**
- **a <u>list</u> of evidence variables $\mathbf{K}$: J, M**
- **a <u>list</u> of observed values k for $\mathbf{K}$: j, m**
- **a list of remaining unobserved variables $\mathbf{Y}$: E, A**

the query can be evaluated as:

$$P(b \mid j, m)$$
$$= \alpha * \sum_{e}\sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let’s change it a bit for simplicity):

$P(Burglary\ =\ true \mid JohnCalls\ =\ true \wedge MaryCalls\ =\ true)$

Given:

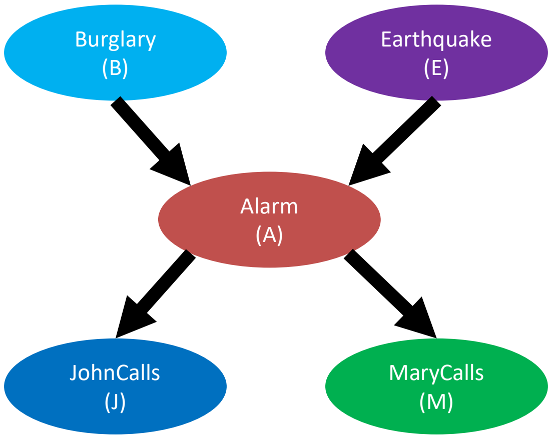
- **a query involving a single variable** B
- **a list of evidence variables** $\mathbf{K}$**:** J, M
- **a list of observed values** $\mathbf{k}$ **for** $\mathbf{K}$**:** j, m
- **a list of remaining unobserved variables** $\mathbf{Y}$**:** E, A

the query can be evaluated as:

$$P(b \mid j, m)$$
$$= \alpha * \sum_{e} \sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \sum_{e} P(e) * \sum_{a} P(a|b,e) * P(j|a) * P(m|a)$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let’s change it a bit for simplicity):

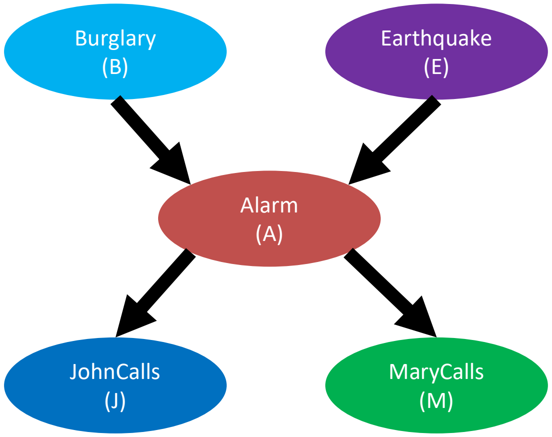
$P(Burglary\ =\ true\ |\ JohnCalls\ =\ true \wedge MaryCalls\ =\ true)$

Query rewritten:

$$P(b\ |\ j, m)$$
$$= \alpha * \sum_{e}\sum_{a} P(b) * P(e) * P(a|b, e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \sum_{e} P(e) * \sum_{a} P(a|b, e) * P(j|a) * P(m|a)$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let's change it a bit for simplicity):

$P(Burglary\ =\ true\ |\ JohnCalls\ =\ true \wedge MaryCalls\ =\ true)$

Query rewritten:

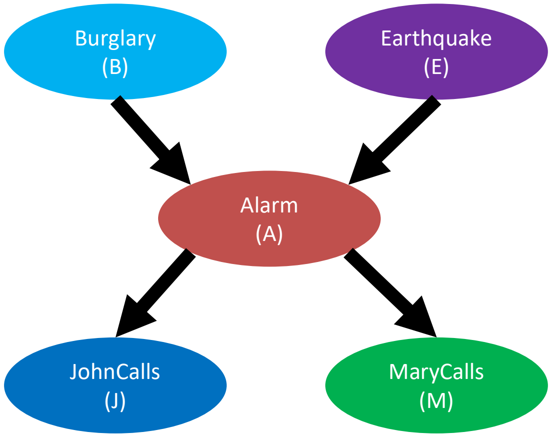
$$P(b\ |\ j, m)$$

$$=\ \alpha * \sum_{e}\sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$

$$=\ \alpha * \boxed{P(b)} * \sum_{e} P(e) * \sum_{a} P(a|b,e) * P(j|a) * P(m|a)$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

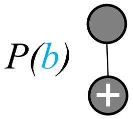
Inference

Query (let’s change it a bit for simplicity):

$$P(Burglary\ =\ true\ |\ JohnCalls\ =\ true \wedge MaryCalls\ =\ true)$$

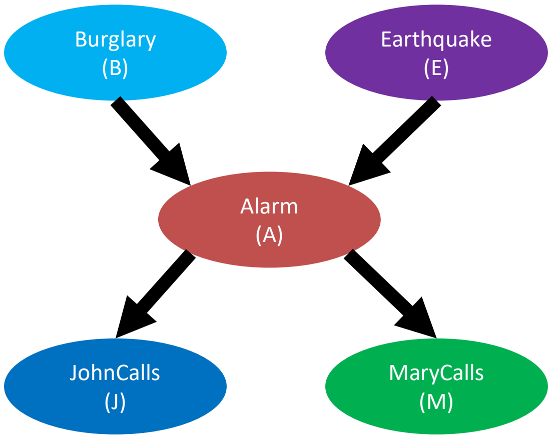
Query rewritten:

$$P(b\ |\ j, m)$$
$$= \alpha * \sum_e \sum_a P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \boxed{\sum_e} P(e) * \sum_a P(a|b,e) * P(j|a) * P(m|a)$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

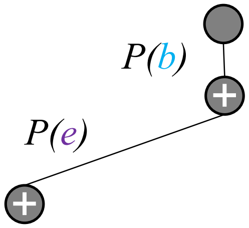
Inference

Query (let's change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

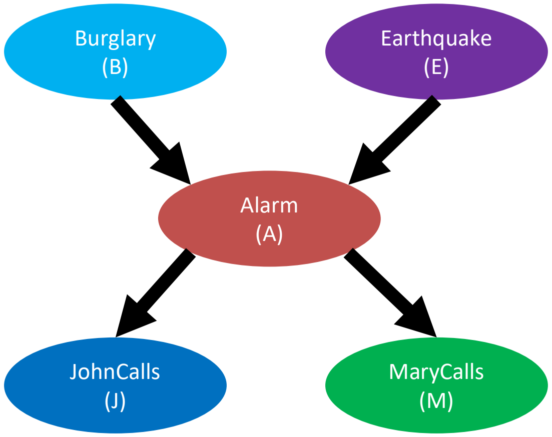
Query rewritten:

$$P(b \mid j, m)$$
$$= \alpha * \sum_{e} \sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \boxed{\sum_{e} P(e) *} \sum_{a} P(a|b,e) * P(j|a) * P(m|a)$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let’s change it a bit for simplicity):

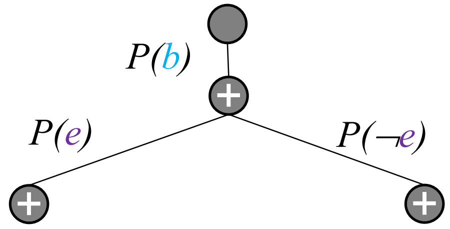
$P(Burglary = true \mid JohnCalls = true \land MaryCalls = true)$

Query rewritten:

$$P(b \mid j, m)$$

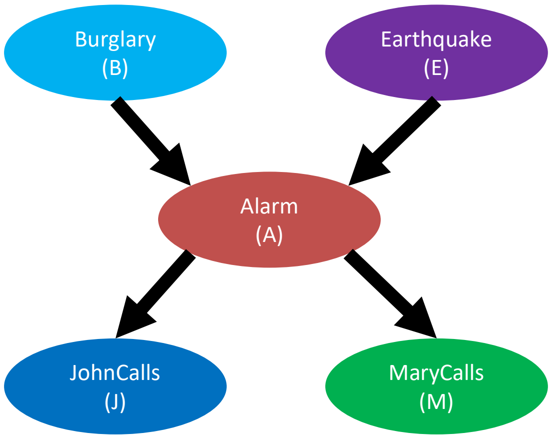
$$= \alpha * \sum_e \sum_a P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$

$$= \alpha * P(b) * \boxed{\sum_e P(e) *} \sum_a P(a|b,e) * P(j|a) * P(m|a)$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

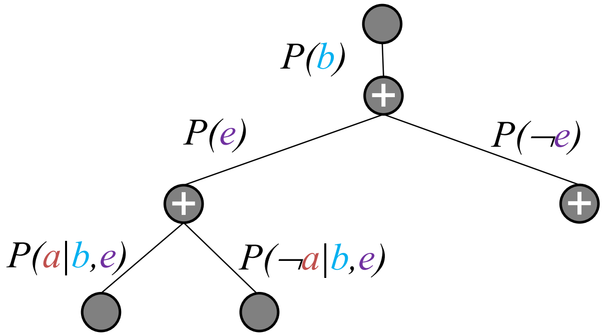
Inference

Query (let's change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

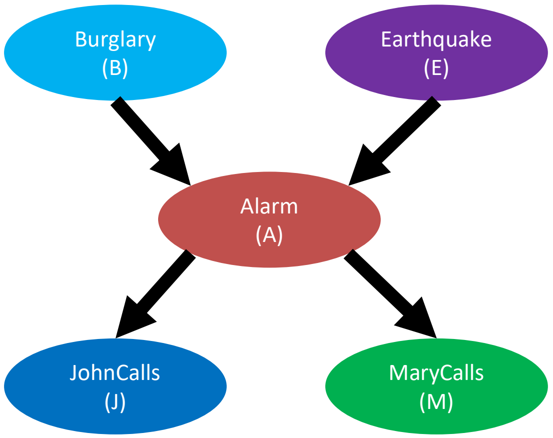
Query rewritten:

$$
\begin{aligned}
&P(b \mid j, m) \\
&= \alpha * \sum_e \sum_a P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a) \\
&= \alpha * P(b) * \sum_e P(e) * \boxed{\sum_a \boxed{P(a|b,e)} * P(j|a) * P(m|a)}
\end{aligned}
$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

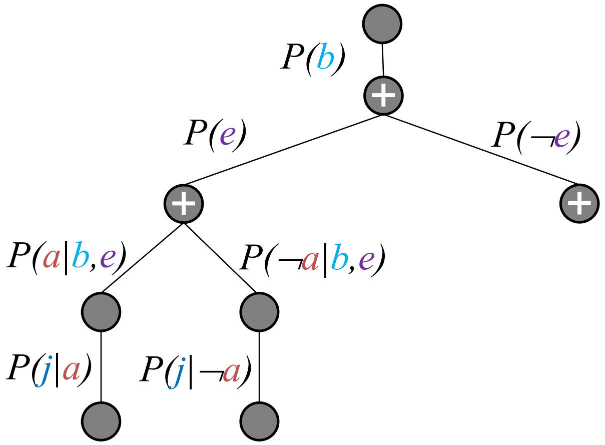
Inference

Query (let's change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

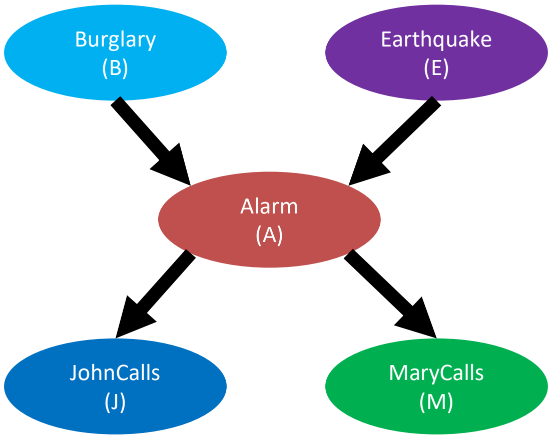
Query rewritten:

$$P(b \mid j, m)$$
$$= \alpha * \sum_e \sum_a P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \sum_e P(e) * \boxed{\sum_a P(a|b,e) * \boxed{P(j|a)} * P(m|a)}$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

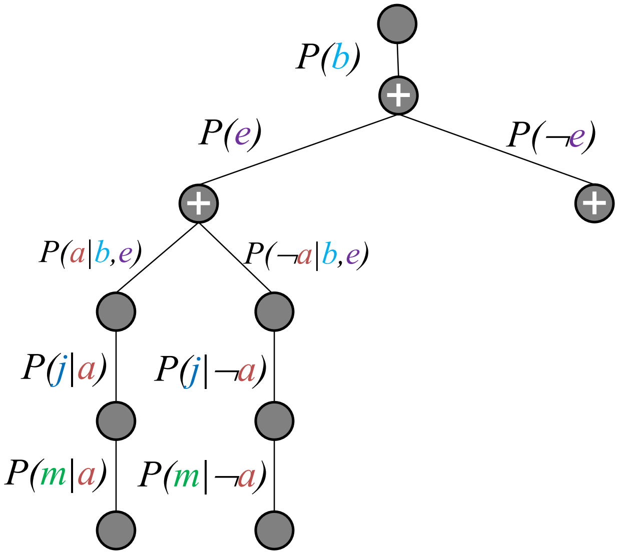
Inference

Query (let's change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

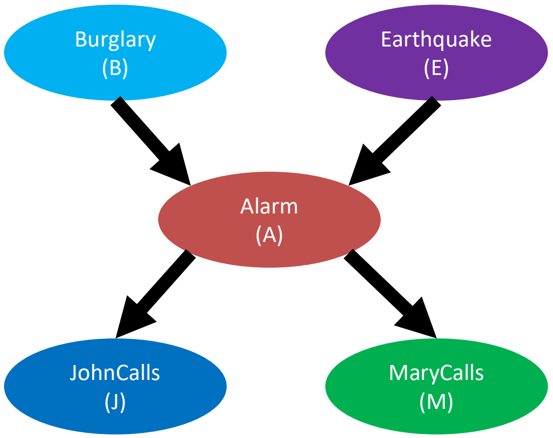
Query rewritten:

$$P(b \mid j, m)$$
$$= \alpha * \sum_{e} \sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * P(b) * \sum_{e} P(e) * \boxed{\sum_{a} P(a|b,e) * P(j|a) * \boxed{P(m|a)}}$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

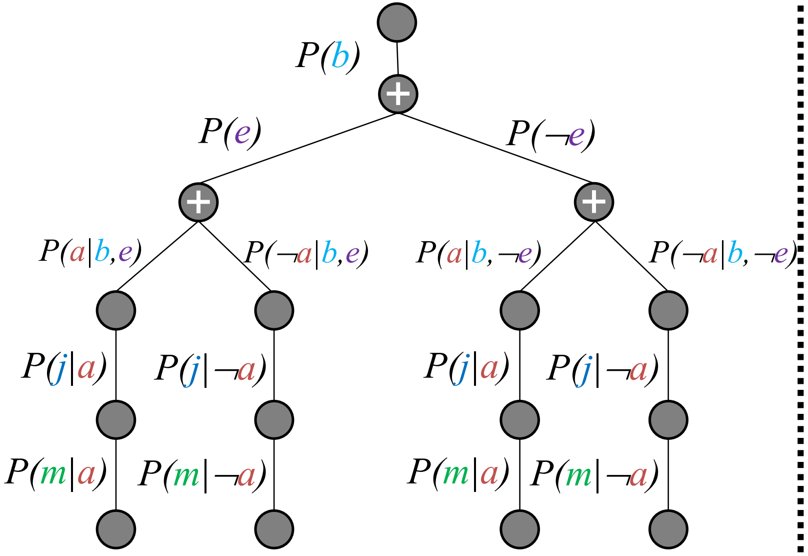
Inference

Query (let’s change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

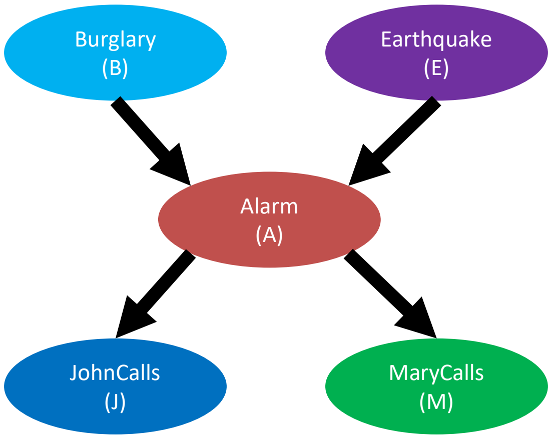
Query rewritten:

$$P(b \mid j, m) = \alpha * \sum_{e}\sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a) = \alpha * P(b) * \sum_{e} P(e) * \boxed{\sum_{a} P(a|b,e) * P(j|a) * P(m|a)}$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

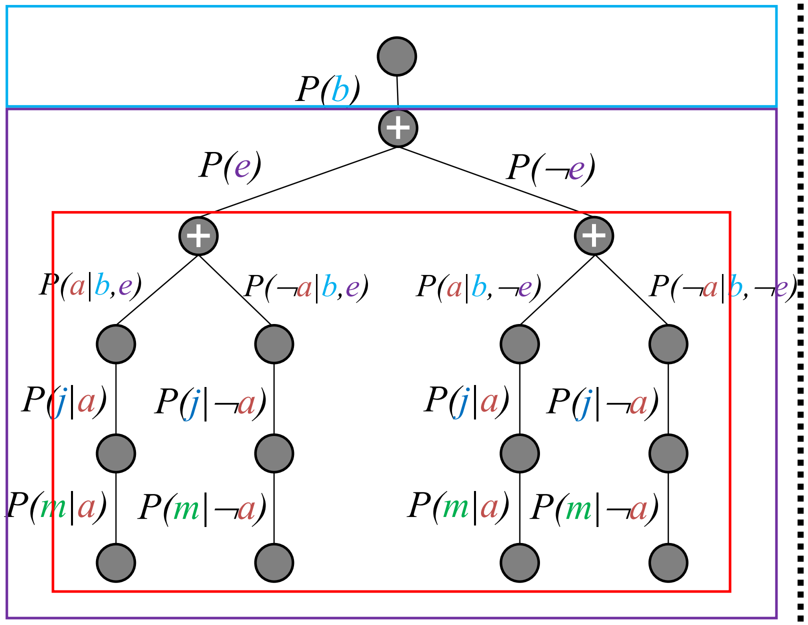
Inference

Query (let’s change it a bit for simplicity):

$P(Burglary = true \mid JohnCalls = true \wedge MaryCalls = true)$

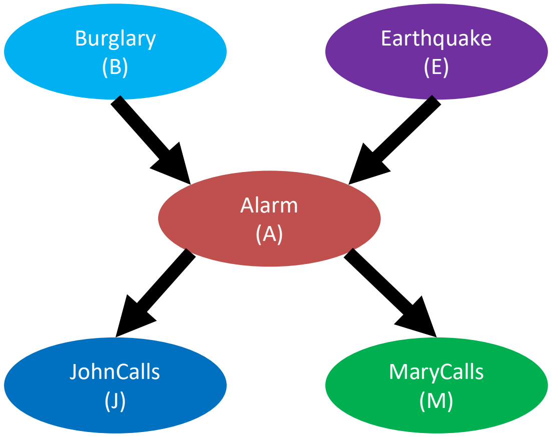
Query rewritten:

$$P(b \mid j, m)$$
$$= \alpha * \sum_e \sum_a P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$
$$= \alpha * \boxed{P(b)} * \boxed{\sum_e P(e) * \boxed{\sum_a P(a|b,e) * P(j|a) * P(m|a)}}$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let's change it a bit for simplicity):

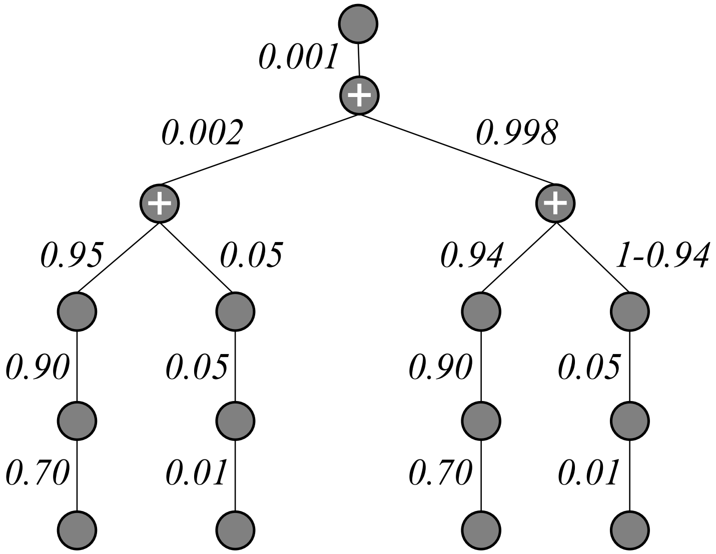
$$P(Burglary = true \mid JohnCalls = true \land MaryCalls = true)$$

Query rewritten:

$$P(b \mid j, m)$$

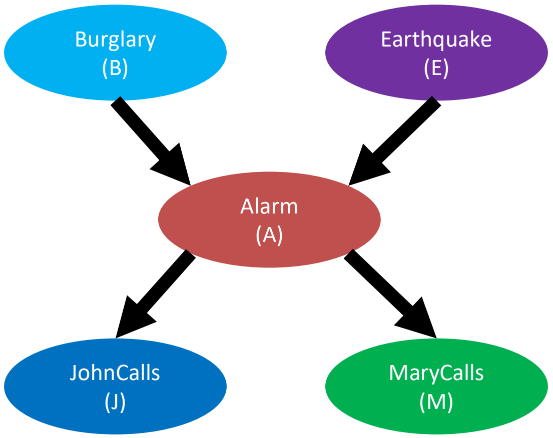
$$= \alpha * \sum_{e} \sum_{a} P(b) * P(e) * P(a|b,e) * P(j|a) * P(m|a)$$

$$= \alpha * P(b) * \sum_{e} P(e) * \sum_{a} P(a|b,e) * P(j|a) * P(m|a)$$



P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (let’s change it a bit for simplicity):

$P(Burglary\ =\ true\ |\ JohnCalls\ =\ true \wedge MaryCalls\ =\ true)$

We can now calculate:

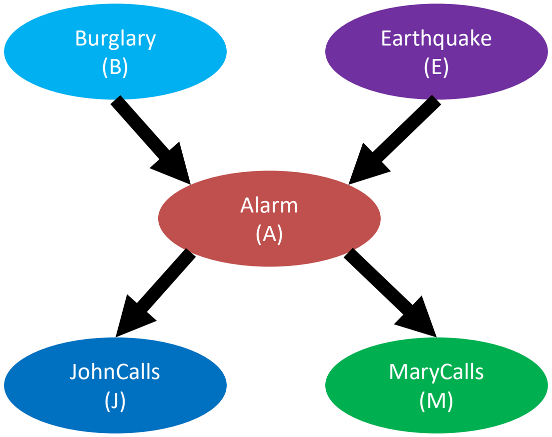
$$P(b\ |\ j, m) = \alpha * 0.00059224$$

And through similar process (not shown):

$$P(\neg b\ |\ j, m) = \alpha * 0.0014919$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (now we can get joint distribution):

$$\boldsymbol{P}(Burglary \mid JohnCalls = true \wedge MaryCalls = true)$$

We can now calculate:

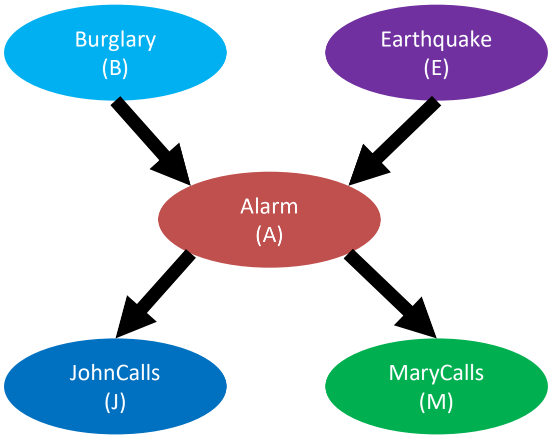
$$\boldsymbol{P}(B \mid j, m) = \alpha * \ < 0.00059224, 0.0014919 >$$

And after normalization:

$$\boldsymbol{P}(B \mid j, m) \approx < 0.284, 0.716 >$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Inference

Query (now we can get joint distribution):

$$\boldsymbol{P}(Burglary \mid JohnCalls = true \land MaryCalls = true)$$

We can now calculate:

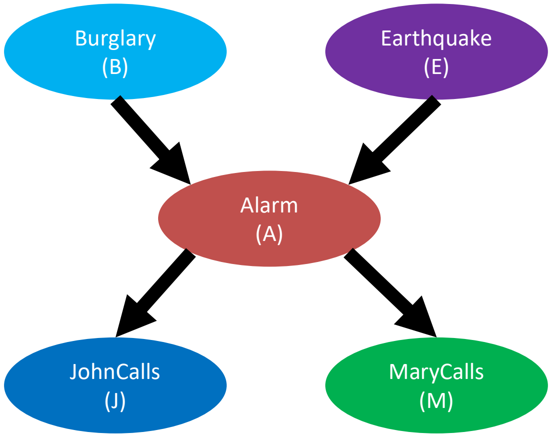
$$\boldsymbol{P}(B \mid j, m) = \alpha * \; < 0.00059224, 0.0014919 >$$

And after normalization:

$$\boldsymbol{P}(B \mid j, m) \approx < 0.284, 0.716 >$$

P(B)	P(¬B)
0.001	0.999

P(E)	P(¬E)
0.002	0.998



B	E	P(A|B,E)
t	t	0.95
t	f	0.94
f	t	0.29
f	f	0.001

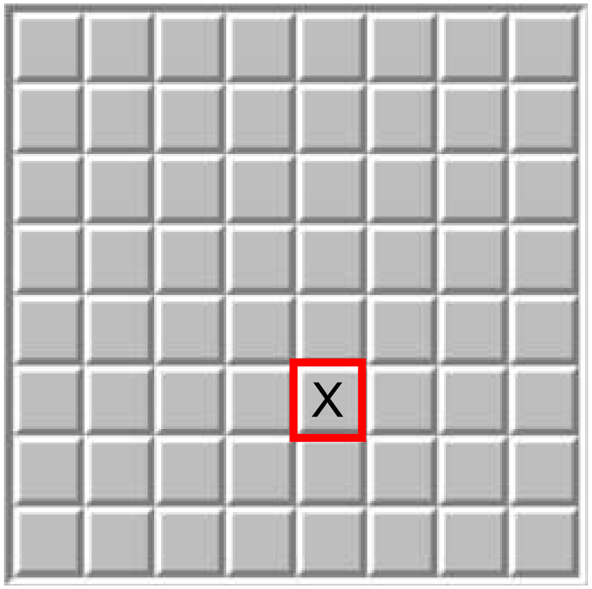
A	P(J|A)
t	0.90
f	0.05

A	P(M|A)
t	0.70
f	0.01

Playing Minesweeper with Bayes’ Rule

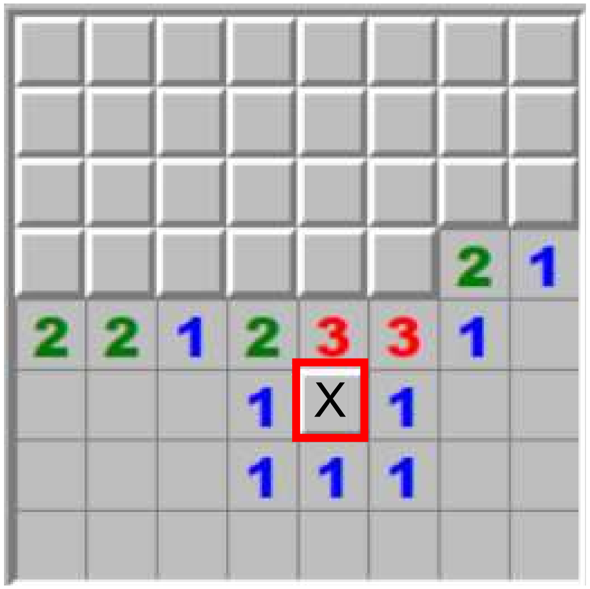
Prior probability / belief:

$$P(X = mine) = 0.5$$

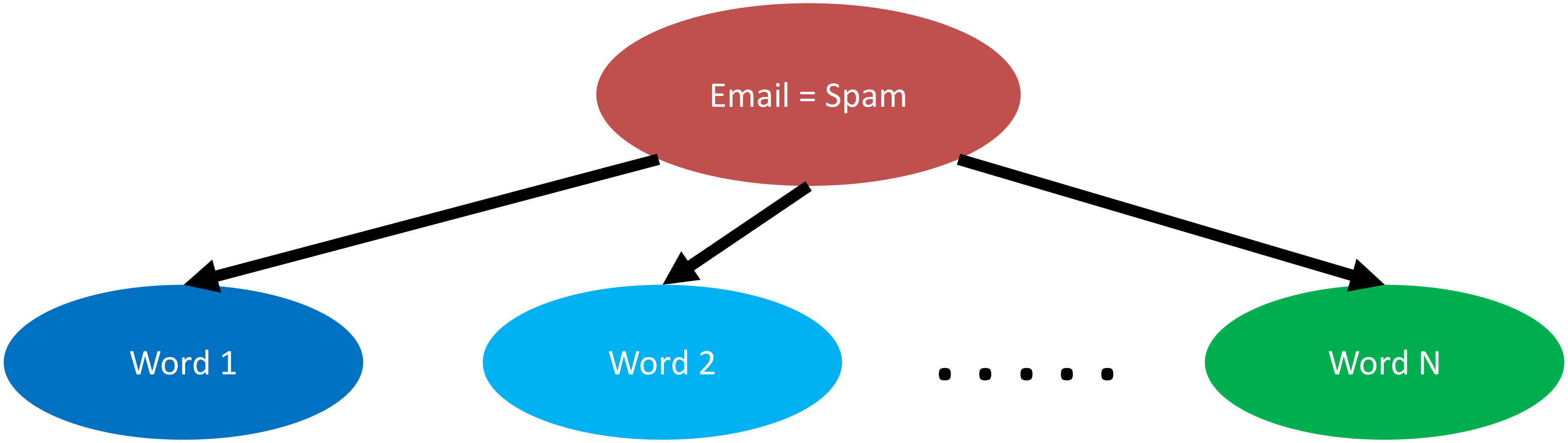


Posterior probability / belief:

$$P(X = mine \mid \boldsymbol{evidence}) = 1.0$$



Naive Bayes Spam Filter



$P(\text{Email} = \text{spam} \mid \text{Word1}) = 0.09$

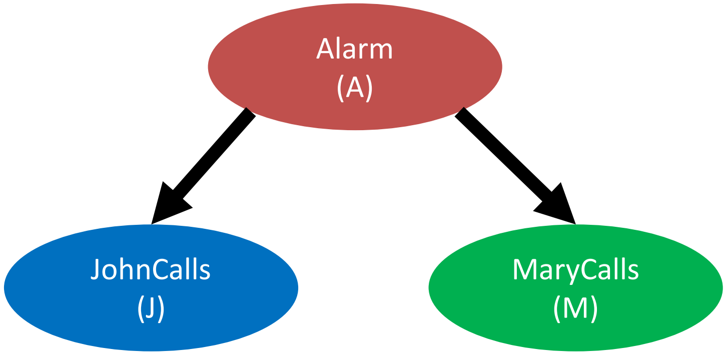
$P(\text{Email} = \text{spam} \mid \text{Word2}) = 0.01$

…

$P(\text{Email} = \text{spam} \mid \text{WordN}) = 0.03$

Conditional Independence

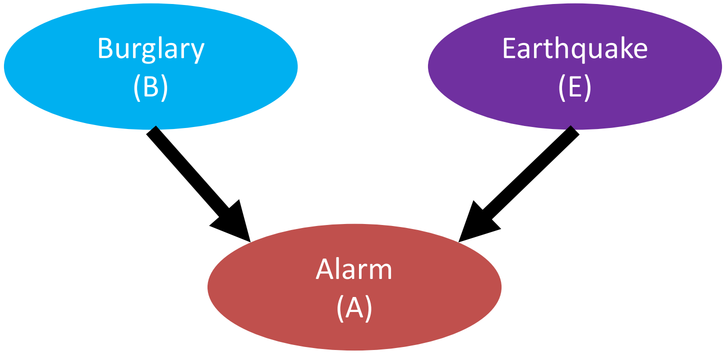
Common Cause:



JohnCalls and MaryCalls are NOT independent

JohnCalls and MaryCalls are CONDITIONALLY independent given Alarm

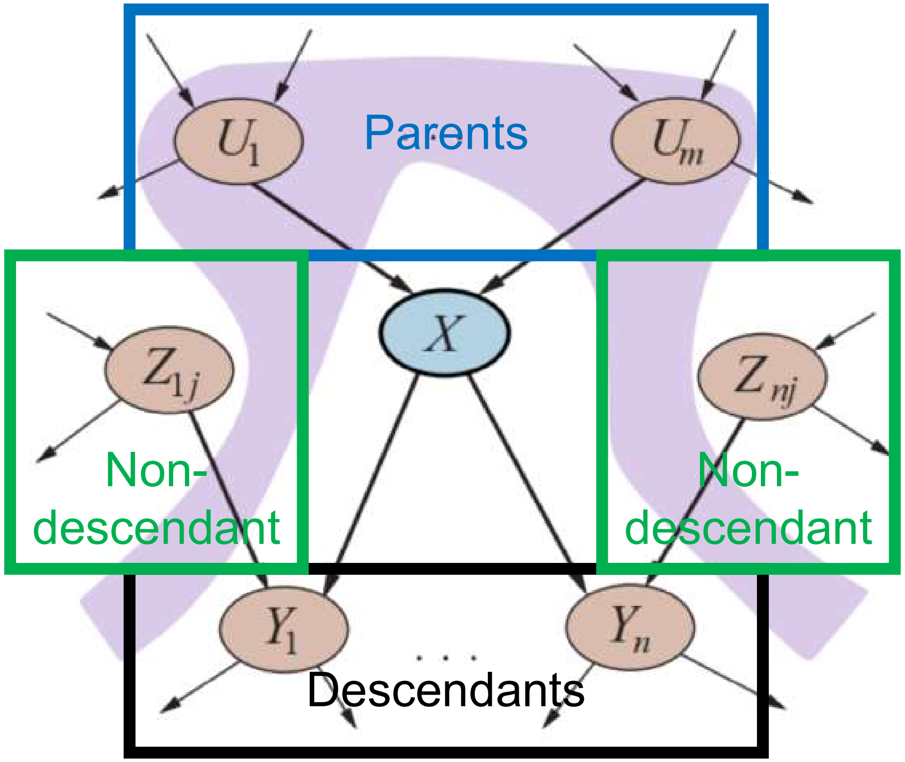
Common Effect:



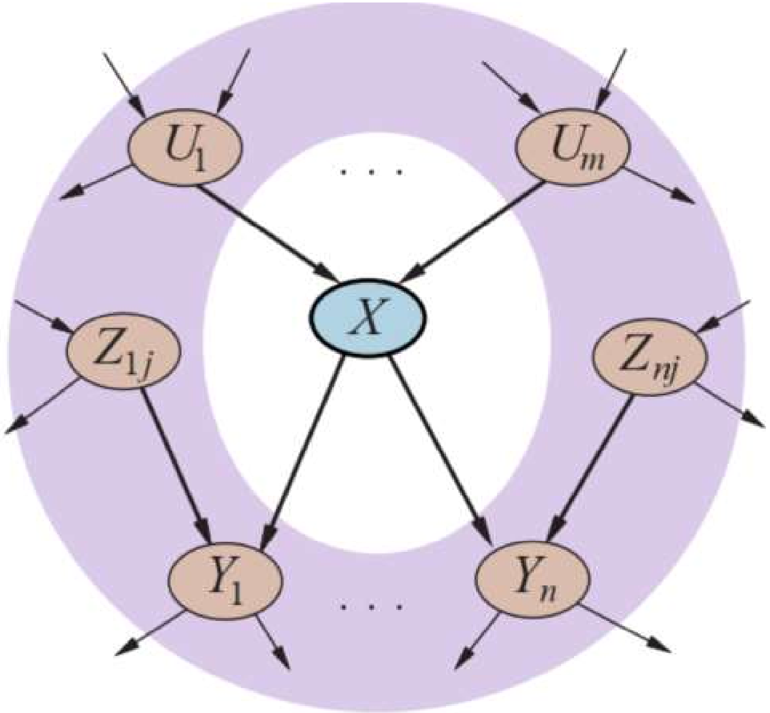
Burglary and Earthquake are independent

Burglary and Earthquake are NOT CONDITIONALLY independent given Alarm

Conditional Independence



Node X is conditionally independent of its non-descendants given its parents.



Node X is conditionally independent of ALL other nodes in the network its given its Markov blanket.

Why do we care?

An unconstrained joint probability distribution with N **binary** variables involves 2^N probabilities.
Bayesian network with at most k parents per each node (N) involves $N * 2^k$ probabilities ($k < N$).

Chain Rule

Conditional probabilities can be used to decompose conjunctions using the chain rule. For any propositions $\boldsymbol{f_1, f_2, \ldots, f_n}$:

$P(f_1 \wedge f_2 \wedge \ldots \wedge f_n) \; =$

$P(f_1) \; *$

$P(f_2 \mid f_1) \; *$

$P(f_3 \mid f_1 \wedge f_2) \; *$

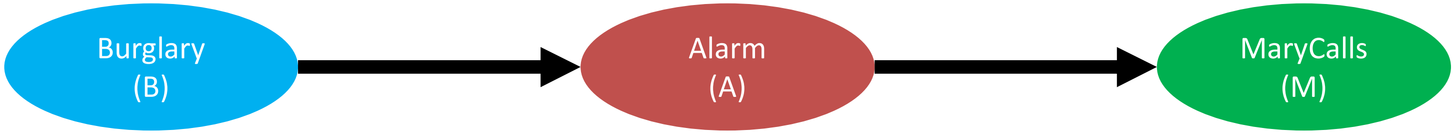
$\ldots$

$P(f_n \mid f_1 \wedge \ldots \wedge f_{n-1}) \; =$

$= \prod_{i=1}^{n} P(f_i \mid Parents(f_i))$ ← Enabled by conditional independence

Conditional Independence

Causal Chain:

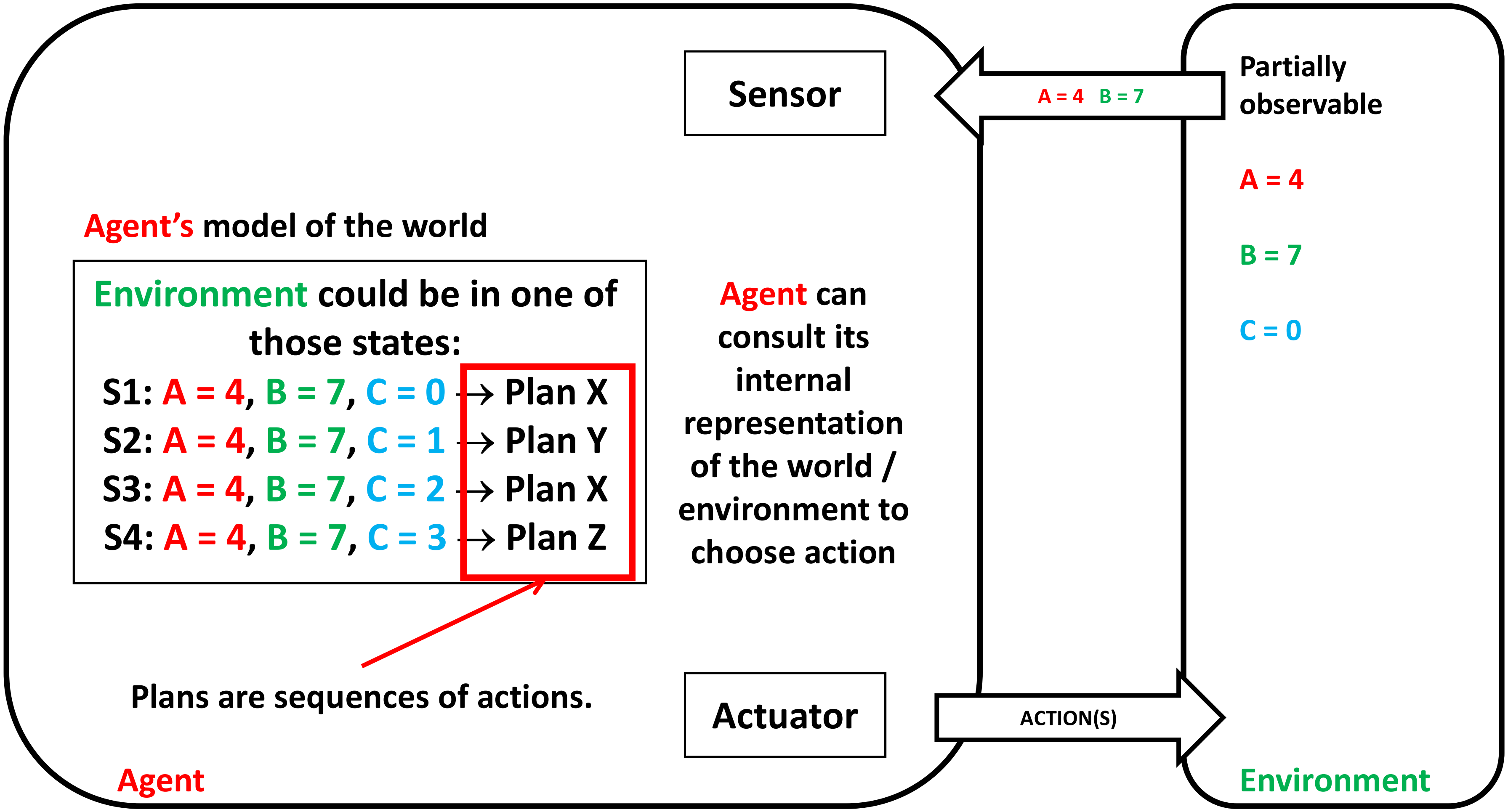


$$P(M \mid A, B) = \frac{P(A, B, M)}{P(A, B)} = \frac{P(B) * P(A \mid B) * P(M \mid A)}{P(B) * P(A \mid B)} = P(M \mid A)$$

Burglary and MaryCalls are CONDITIONALLY independent given Alarm.

If Alarm is given, what “happened before” does not directly influence MaryCalls.

Agents and Belief State



Agent's model of the world

Environment could be in one of those states:

S1: A = 4, B = 7, C = 0 → Plan X
S2: A = 4, B = 7, C = 1 → Plan Y
S3: A = 4, B = 7, C = 2 → Plan X
S4: A = 4, B = 7, C = 3 → Plan Z

Agent can consult its internal representation of the world / environment to choose action

Plans are sequences of actions.

Partially observable

A = 4

B = 7

C = 0

Assume: D_C = {0,1,2,3}

Decision Theory

- Decisions: every plan (actions) leads to an outcome (state)
- Agents have preferences (preferred outcomes)
- Preferences → outcome utilities
- Agents have degrees of belief (probabilities) for actions

Decision theory = probability theory + utility theory

Decision Theory

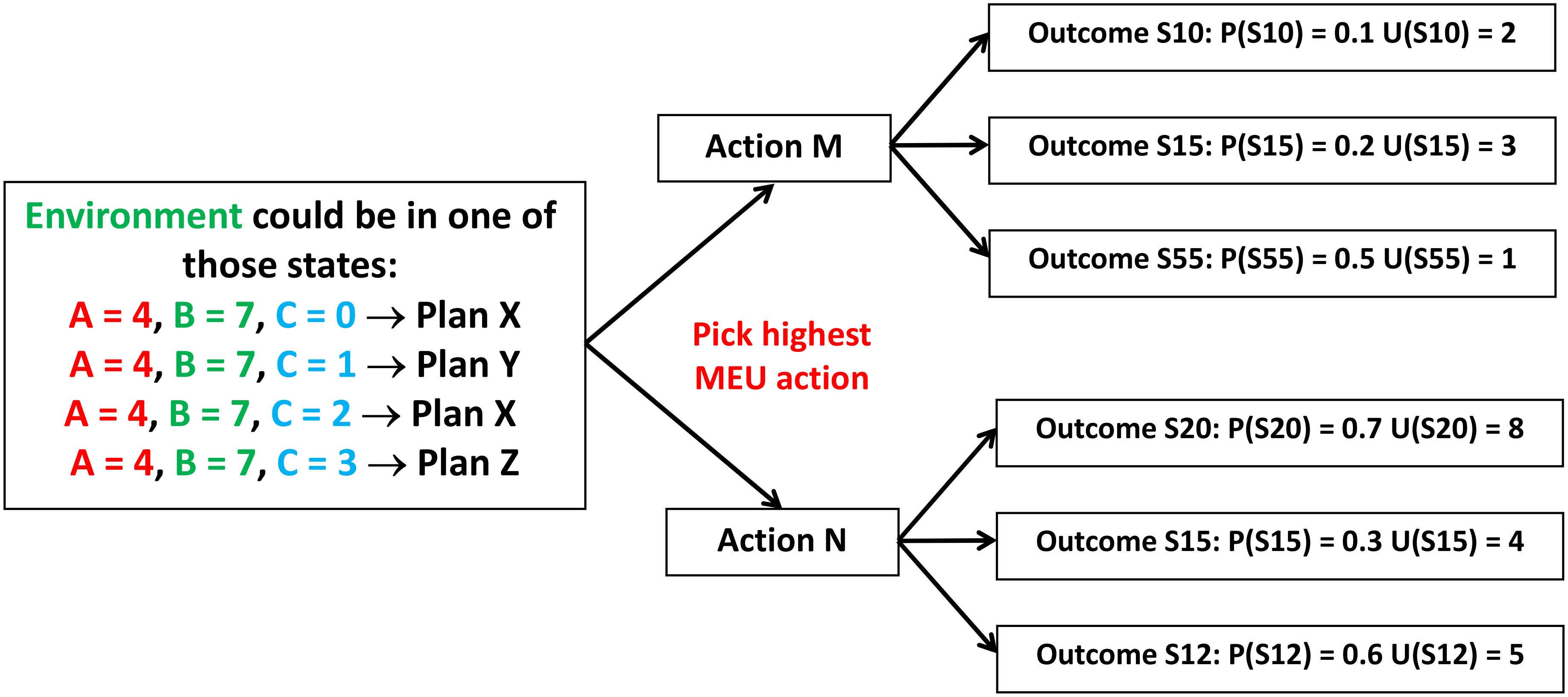
- Decisions: every plan (actions) leads to an outcome (state)
- Agents have preferences (preferred outcomes)
- Preferences → outcome utilities
- Agents have degrees of belief (probabilities) for actions

Decision theory = probability theory + utility theory

BELIEFS DESIRES

Maximum Expected (Average) Utility

$$MEU(M) = P(S10) * U(S10) + P(S15) * U(S15) + P(S55) * U(S55)$$



Environment could be in one of those states:

A = 4, B = 7, C = 0 → Plan X

A = 4, B = 7, C = 1 → Plan Y

A = 4, B = 7, C = 2 → Plan X

A = 4, B = 7, C = 3 → Plan Z

Action M

Outcome S10: P(S10) = 0.1 U(S10) = 2

Outcome S15: P(S15) = 0.2 U(S15) = 3

Outcome S55: P(S55) = 0.5 U(S55) = 1

Pick highest MEU action

Action N

Outcome S20: P(S20) = 0.7 U(S20) = 8

Outcome S15: P(S15) = 0.3 U(S15) = 4

Outcome S12: P(S12) = 0.6 U(S12) = 5

$$MEU(N) = P(S20) * U(S20) + P(S15) * U(S15) + P(S12) * U(S12)$$

Agents Decisions

Recall that agent **ACTIONS** change the state:

- if we are in state s
- action a is expected to
- lead to another state s' (outcome)

Given uncertainty about the current state s and action outcome s' we need to define the following:

- probability (belief) of being in state s: $P(s)$
- probability (belief) of action a leading to outcome s': $P(s' \mid s, a)$

Now:

$$P(s' \mid s, a) = P(\textsc{Result}(a) = s') = \sum_{s} P(s) * P(s' \mid s, a)$$

State Utility Function

Agent's **preferences (desires)** are captured by the **Utility function** $U(s)$.

Utility function assigns a value to each state s to express **how desirable this state is to the agent**.

Expected Action Utility

The expected utility of an action a given the evidence is the average utility value of all possible outcomes s' of action a, weighted by their probability (belief) of occurence:

$$\mathrm{EU}(\mathrm{a}) = \sum_{\mathrm{s'}} \sum_{\mathrm{s}} \mathrm{P}(\mathrm{s}) * \mathrm{P}(\mathrm{s'} \mid \mathrm{s}, \mathrm{a}) * \mathrm{U}(\mathrm{s'}) \quad = \quad \sum_{\mathrm{s'}} \mathrm{P}(Result(a) = s') * \mathrm{U}(\mathrm{s'})$$

Rational agent should choose an action that maximizes the expected utility:

$$\text{chosen action} \;=\; \underset{\mathrm{a}}{\text{argmax}} \; \mathrm{EU}(\mathrm{a})$$

How Did We Get Here?

Let's start with relationships (and related notation) between agent's preferences:

- **agent prefers** A **over** B**:**

  A □ B

- **agent is indifferent between** A **and** B**:**

  A ~ B

- **agent prefers** A **over** B **or is indifferent between** A **and** B **(weak preference):**

  A □ B

The Concept of Lottery

Let's assume the following:

- **an action a is a lottery ticket**
- **the set of outcomes (resulting states) is a lottery**

A lottery L with possible outcomes $S_1, \ldots, S_n$ that occur with probabilities $p_1, \ldots, p_n$ is written as:

$$L = [p_1, S_1; p_2, S_2; \ldots ; p_n, S_n]$$

Lottery outcome S_i: atomic state or another lottery.

Lottery Constraints: Orderability

Given two lotteries A and B, a rational agent must either prefer one or else rate them as equally preferable:

Exactly one of (A □ B), (B □ A), or (A ∼ B) holds

Lottery Constraints: Transitivity

Given three lotteries A**,** B**, and** C**, if an agent prefers** A **to** B **AND prefers** B **to** C**, then the agent must prefer** A **to** C**:**

$$(\mathrm{A}\ \square\ \mathrm{B}) \wedge (\mathrm{B}\ \square\ \mathrm{C}) \Rightarrow (\mathrm{A}\ \square\ \mathrm{C})$$

Lottery Constraints: Continuity

If some lottery B is between A and C in preference, then there is some probability p for which the rational agent will be indifferent between getting B for sure or some other lottery that yields A with probability p and C with probability 1 - p:

$$(A \;\square\; B \;\square\; C) \Rightarrow \exists p\,[p, A; 1- p, C] \sim B$$

Lottery Constraints: Substitutability

If an agent is indifferent between two lotteries A and B, then the agent is indifferent between two more complex lotteries that are the same, except that B is subsituted for A in one of them:

$$(A \sim B) \Rightarrow [p, A; 1 - p, C] \sim [p, B; 1 - p, C]$$

Lottery Constraints: Monotonicity

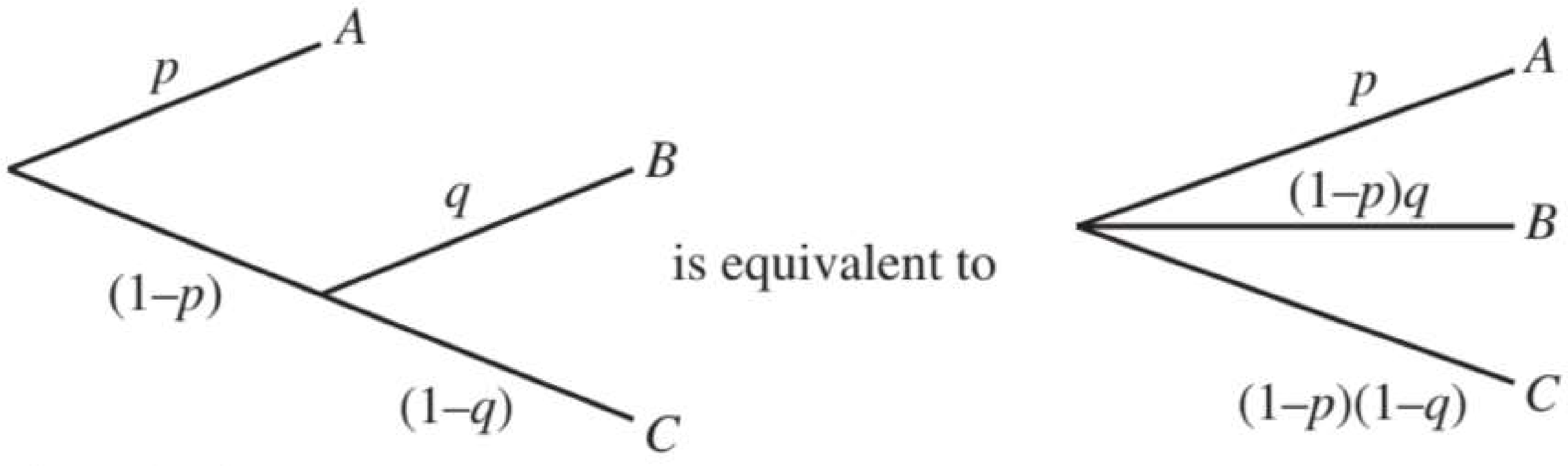
Suppose two lotteries have the same two possible outcomes, A and B. If an agent prefers A to B, then the agent must prefer the lottery that has a higher probability for A:

$$(A \ \square \ B) \Rightarrow (p > q \Leftrightarrow [p, A; 1- p, B] \ \square \ [q, A; 1- q, B])$$

Lottery Constraints: Decomposability

Compound lotteries can be reduced to smaller ones using the laws of probability:

[p, A; 1- p, [q, B; 1- q, C]] ~ [p, A; (1- p)*q, B; (1- p)*(1- q), C]



Preferences and Utility Function

An agent whose preferences between lotteries follow the set of axioms (of utility theory) below:

- **Orderability**
- **Transitivity**
- **Continuity**
- **Subsitutability**
- **Monotonicity**
- **Decomposability**

can be described as possesing a utility function and maximize it.

Preferences and Utility Function

If an agent’s preferences obey the axioms of utility theory, then there exist a function U such that:

$$U(A) = U(B) \text{ **if and only if** } (A \sim B)$$

and

$$U(A) > U(B) \text{ **if and only if** } (A \;\square\; B)$$

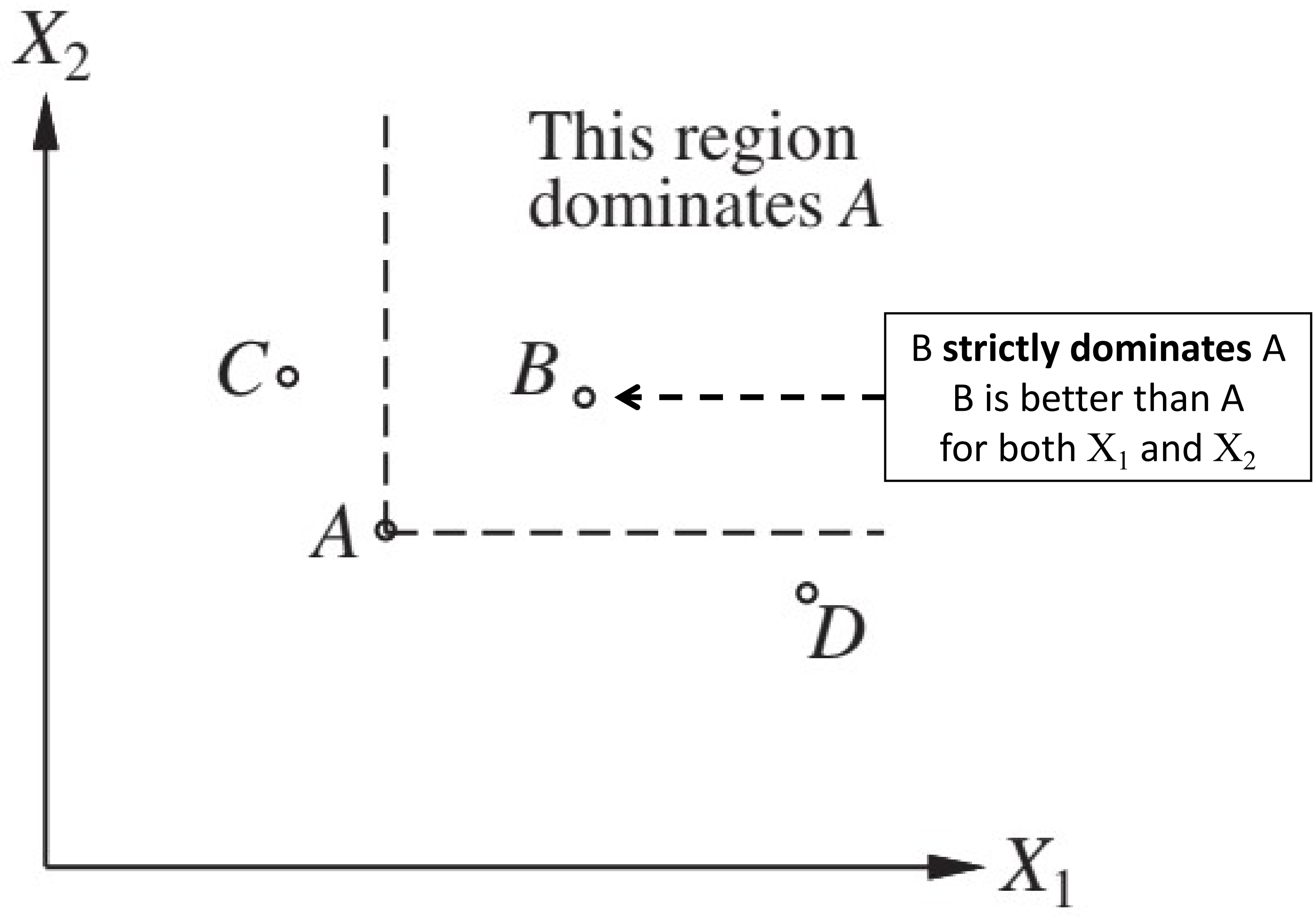
Multiattribute Outcomes

Outcomes can be characterized by more than one attribute. Decisions in such cases are handled by Multiattribute Utility Theory.

Attributes: $\mathbf{X} = X_1, \ldots, X_n$

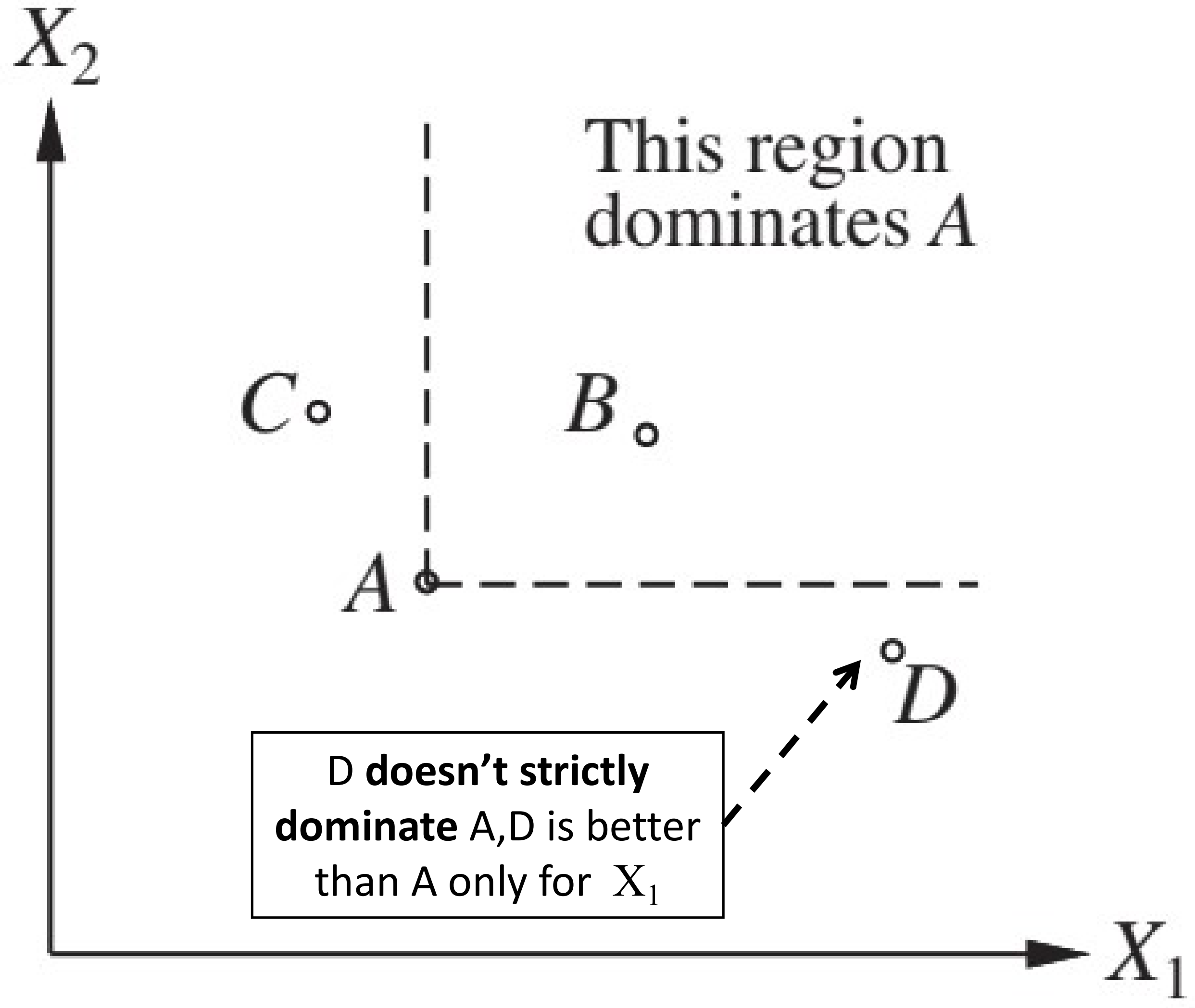
Assigned values: $\mathbf{x} = <x_1, \ldots, x_n>$

Strict Dominance: Deterministic



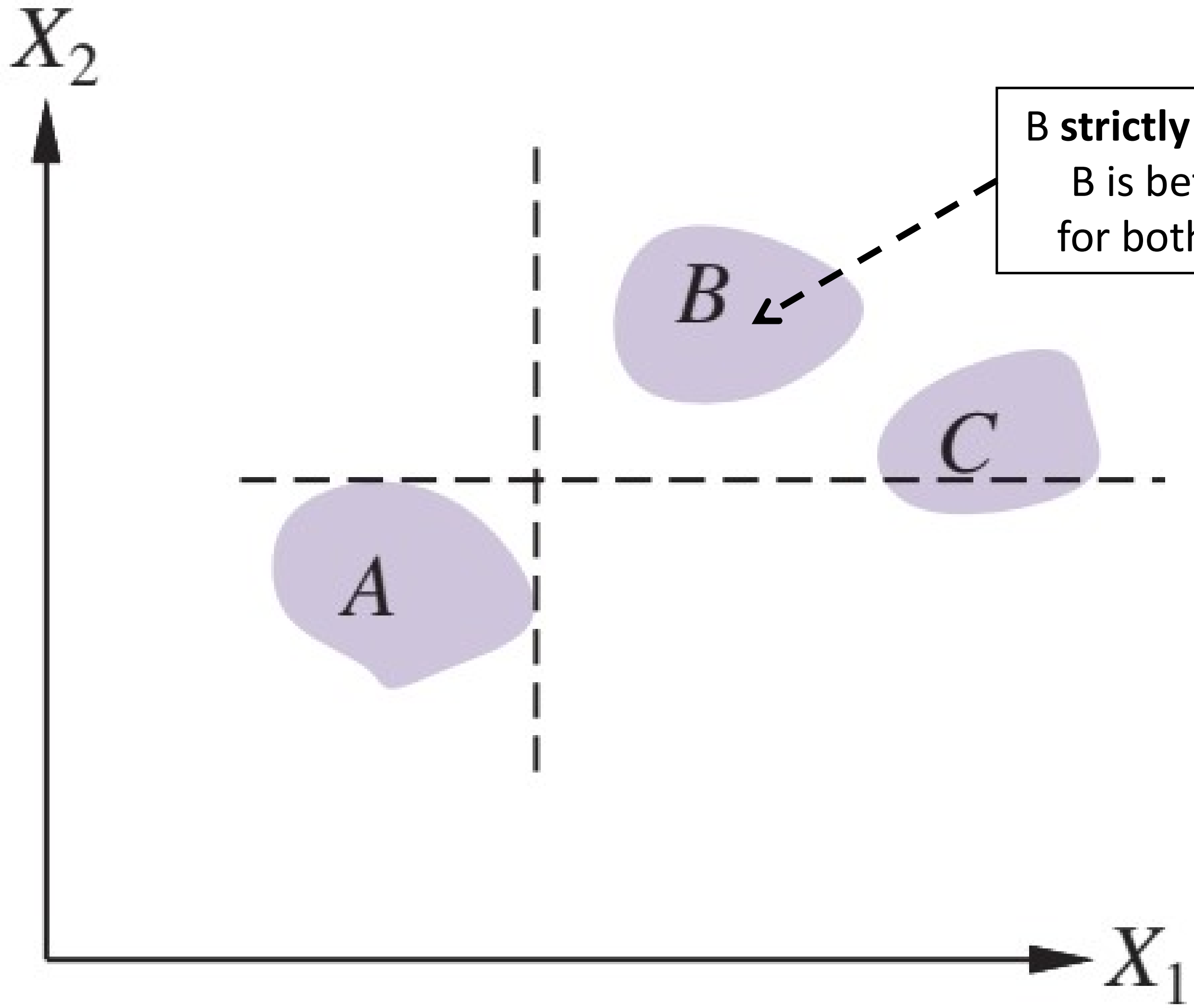
B **strictly dominates** A
B is better than A
for both X_1 and X_2

Strict Dominance: Deterministic



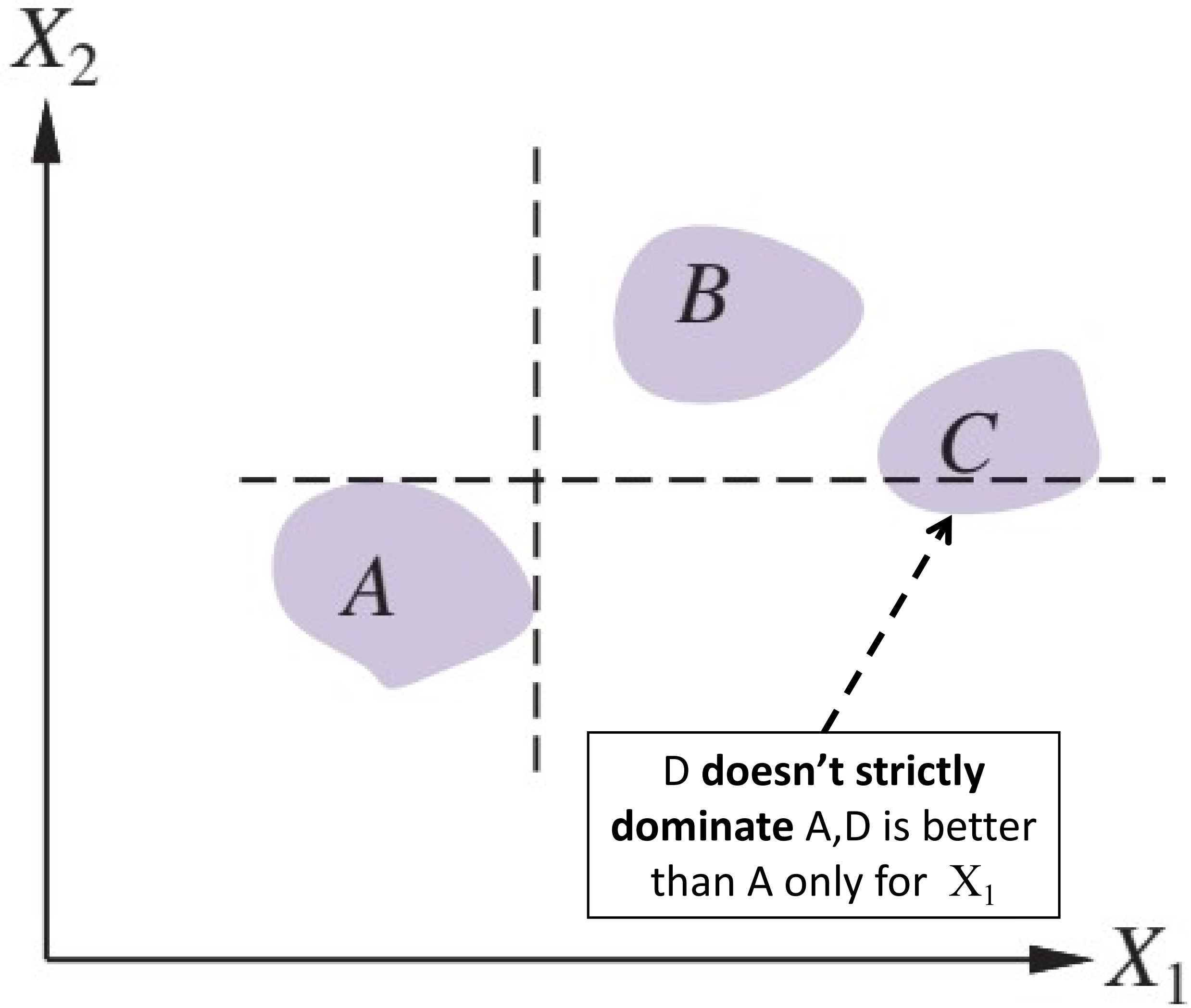
D **doesn't strictly dominate** A,D is better than A only for X_1

Strict Dominance: Uncertain



B **strictly dominates** A
B is better than A
for both X_1 and X_2

Strict Dominance: Uncertain



D **doesn’t strictly dominate** A,D is better than A only for X_1

Decision Network (Influence Diagram)

Decision networks (also called influence diagrams) are structures / mechanisms for making rational decisions.

Decision networks are based on Bayesian networks, but include additional nodes that represent actions and utilities.

Decision Networks

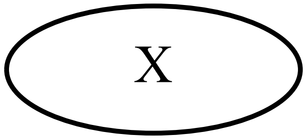
The most basic decision network needs to include:

- information about current state s
- possible actions
- resulting state s' (after applying chosen action a)
- utility of the resulting state $U(s')$

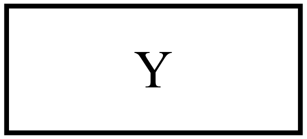
Decision Network Nodes

Decision networks are built using the following nodes:

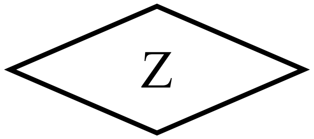
- chance nodes:



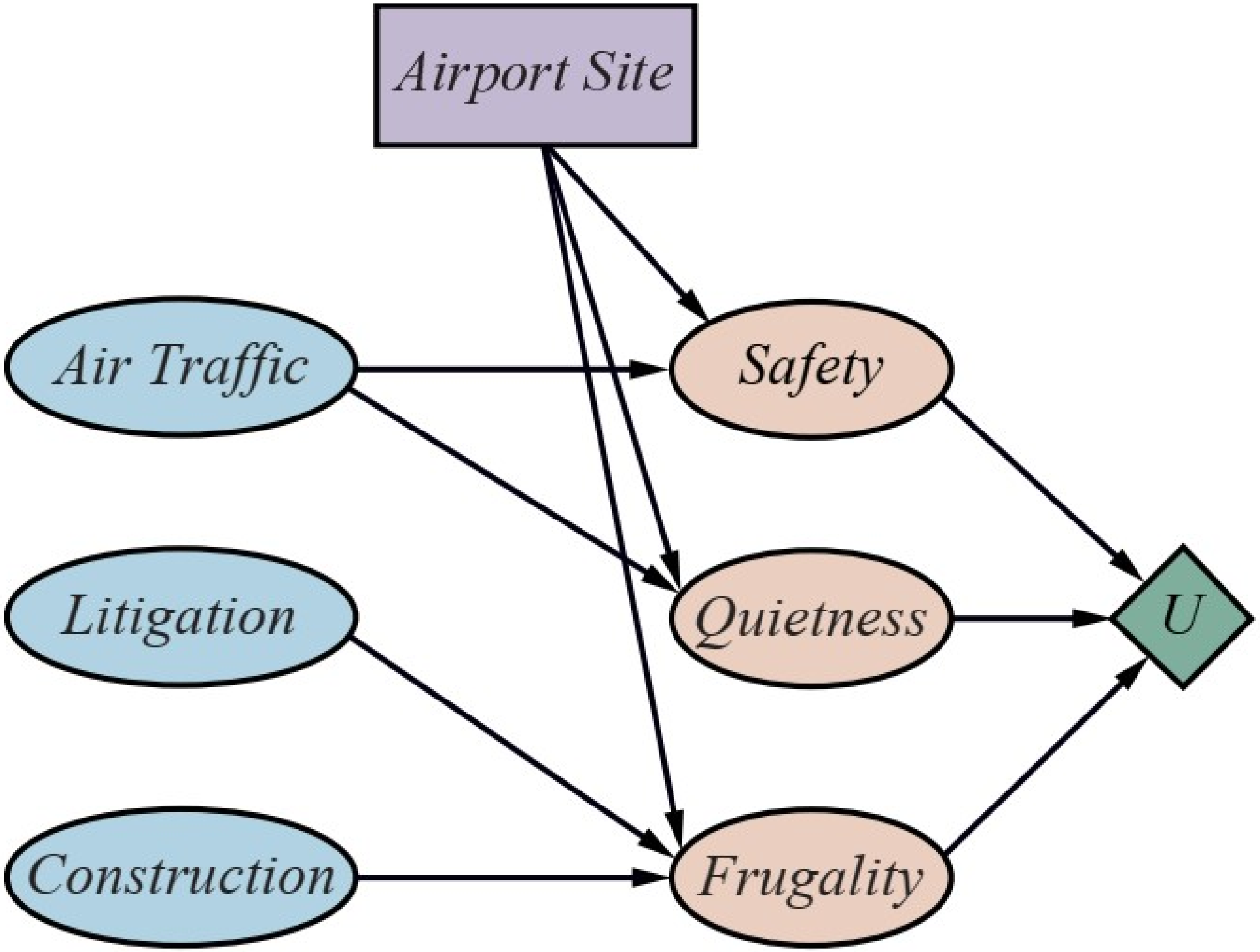
- decision nodes:



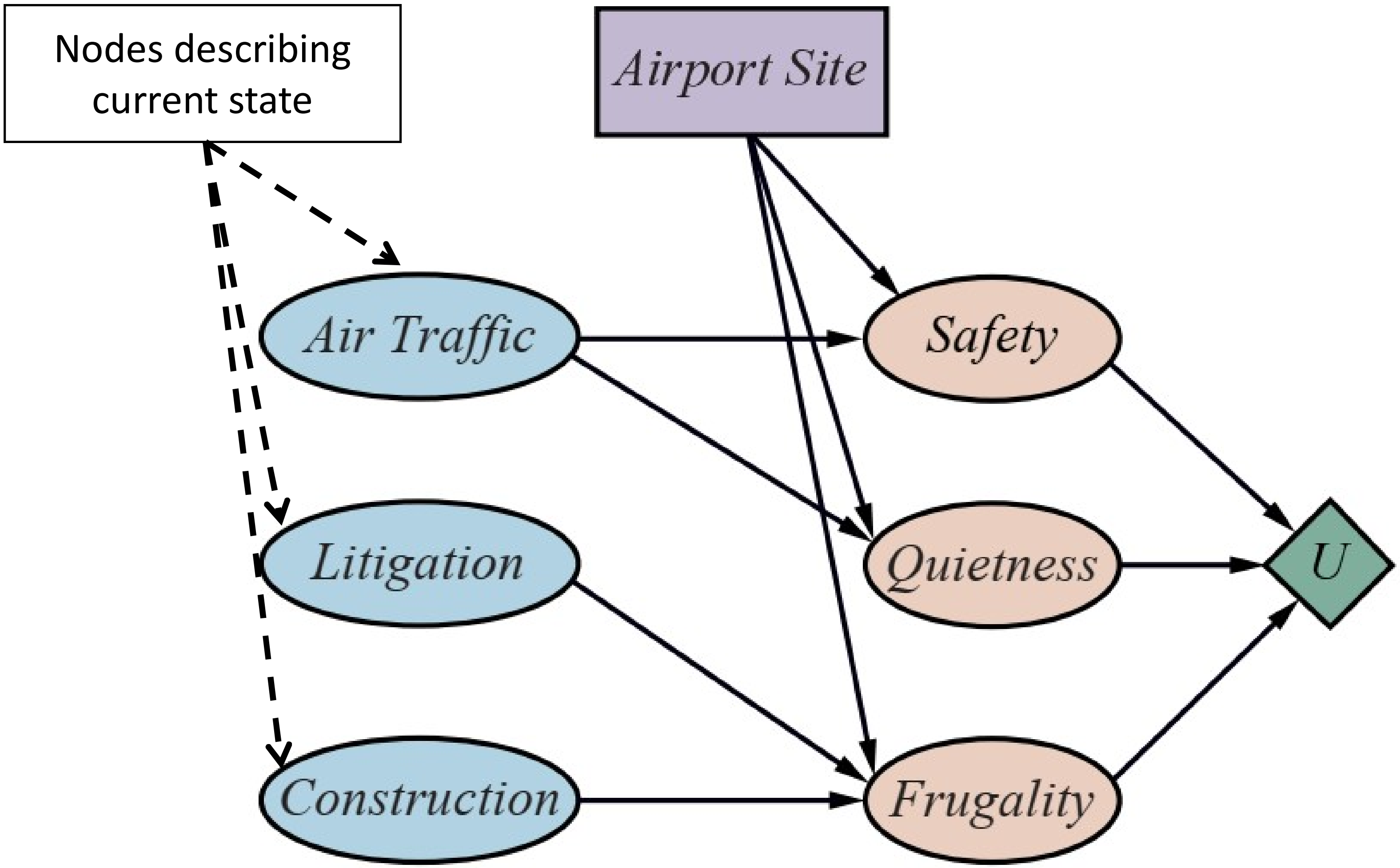
- utility (or value) nodes



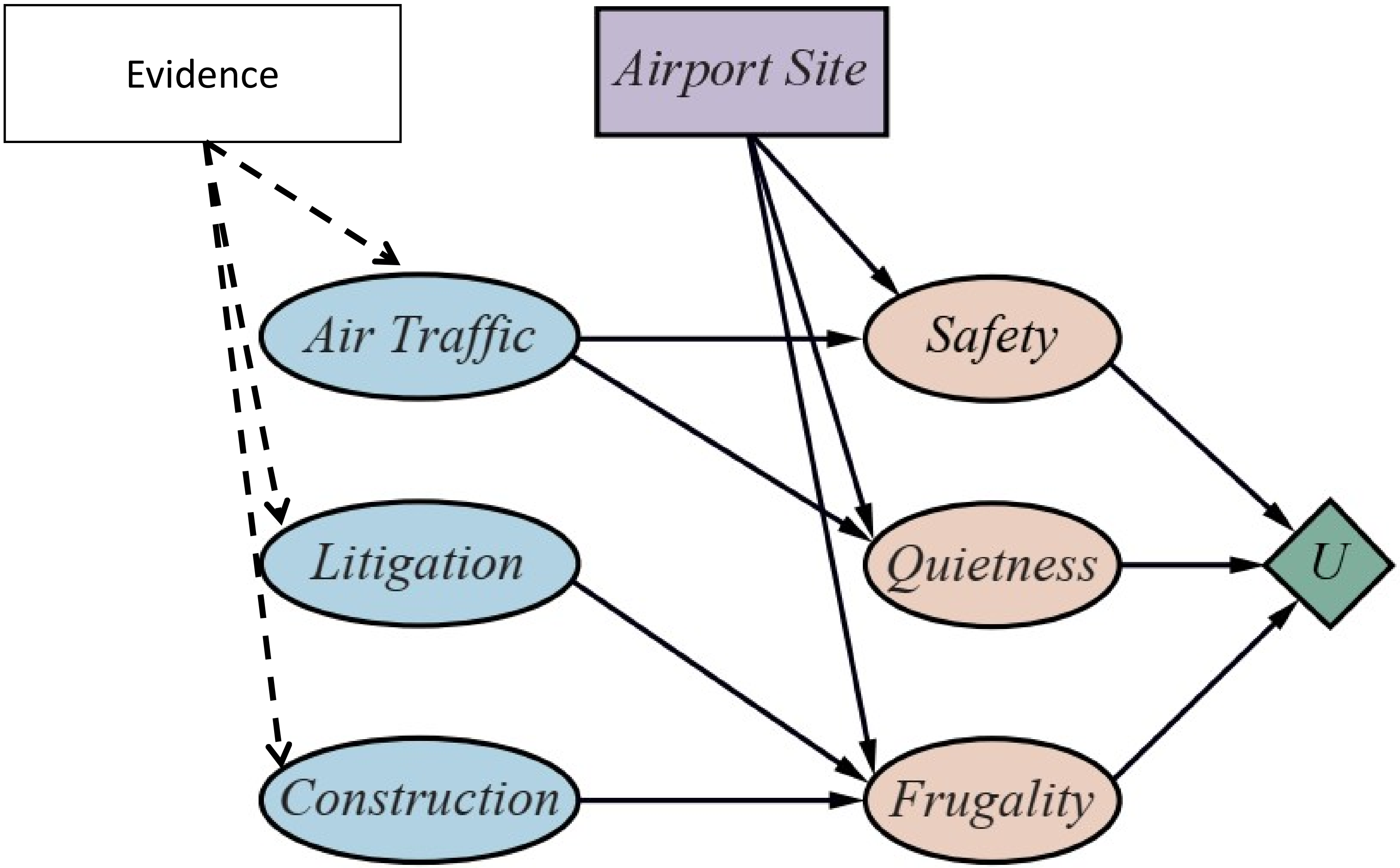
Decision Network: Example



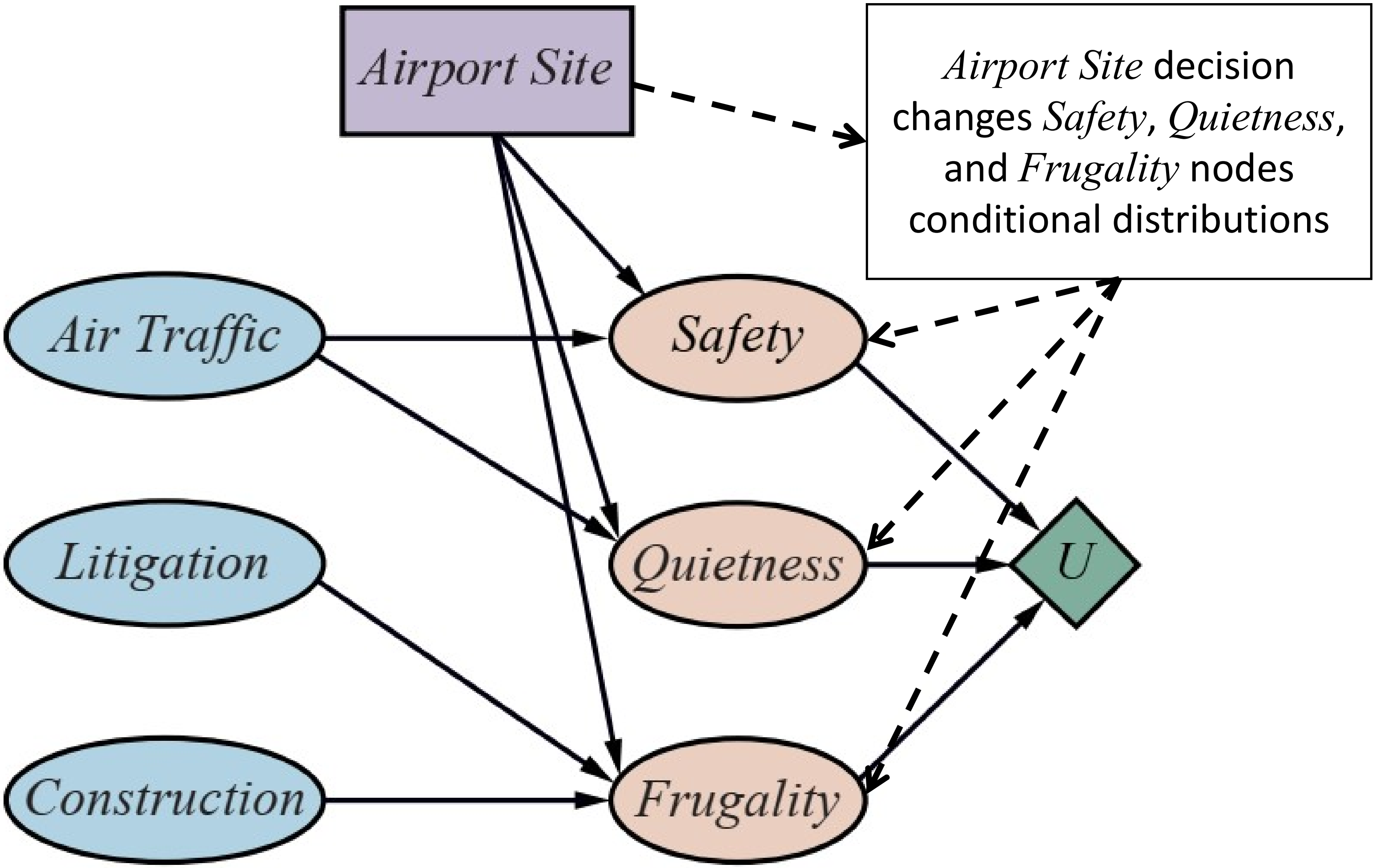
Decision Network: Example



Decision Network: Example

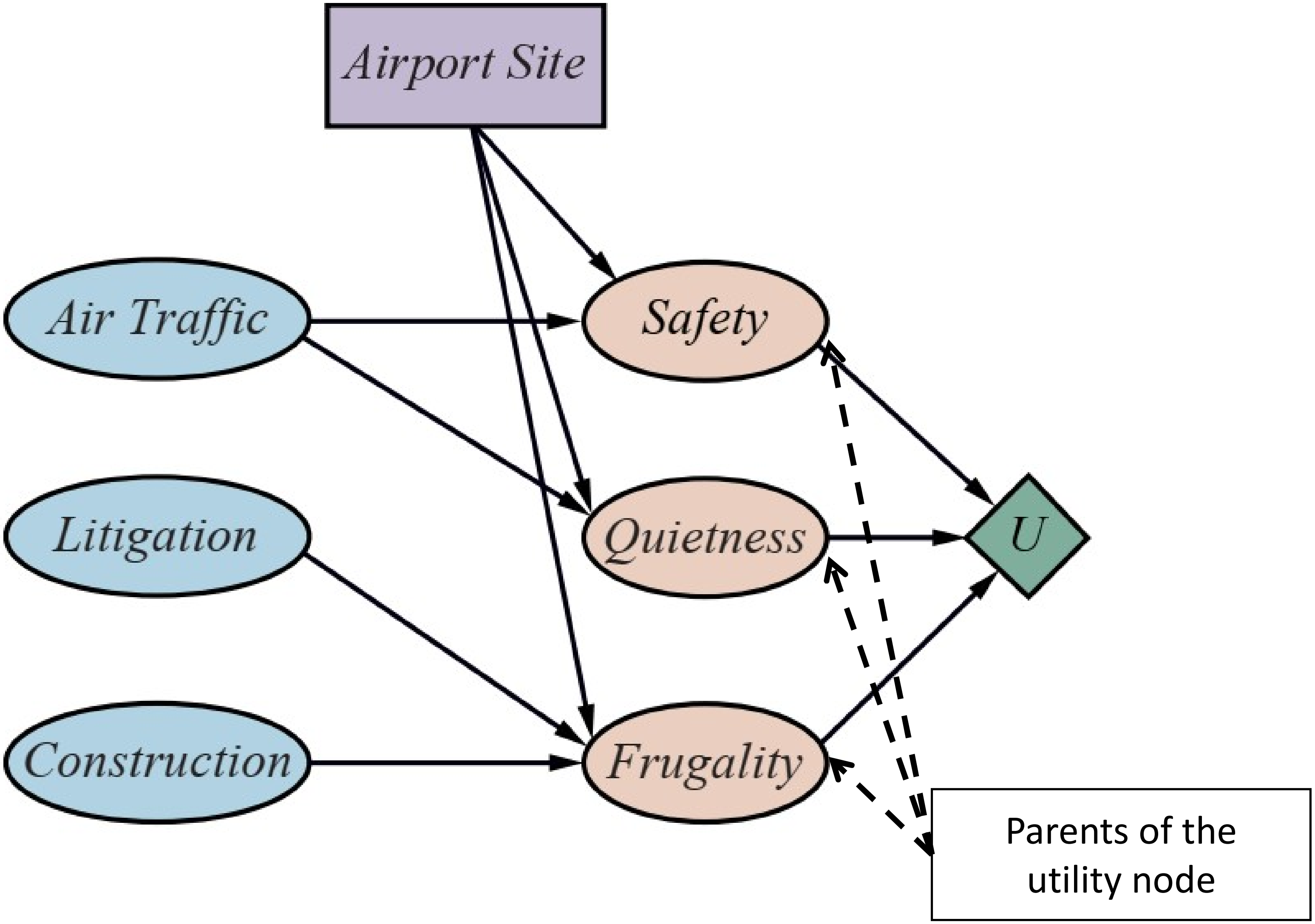


Decision Network: Example



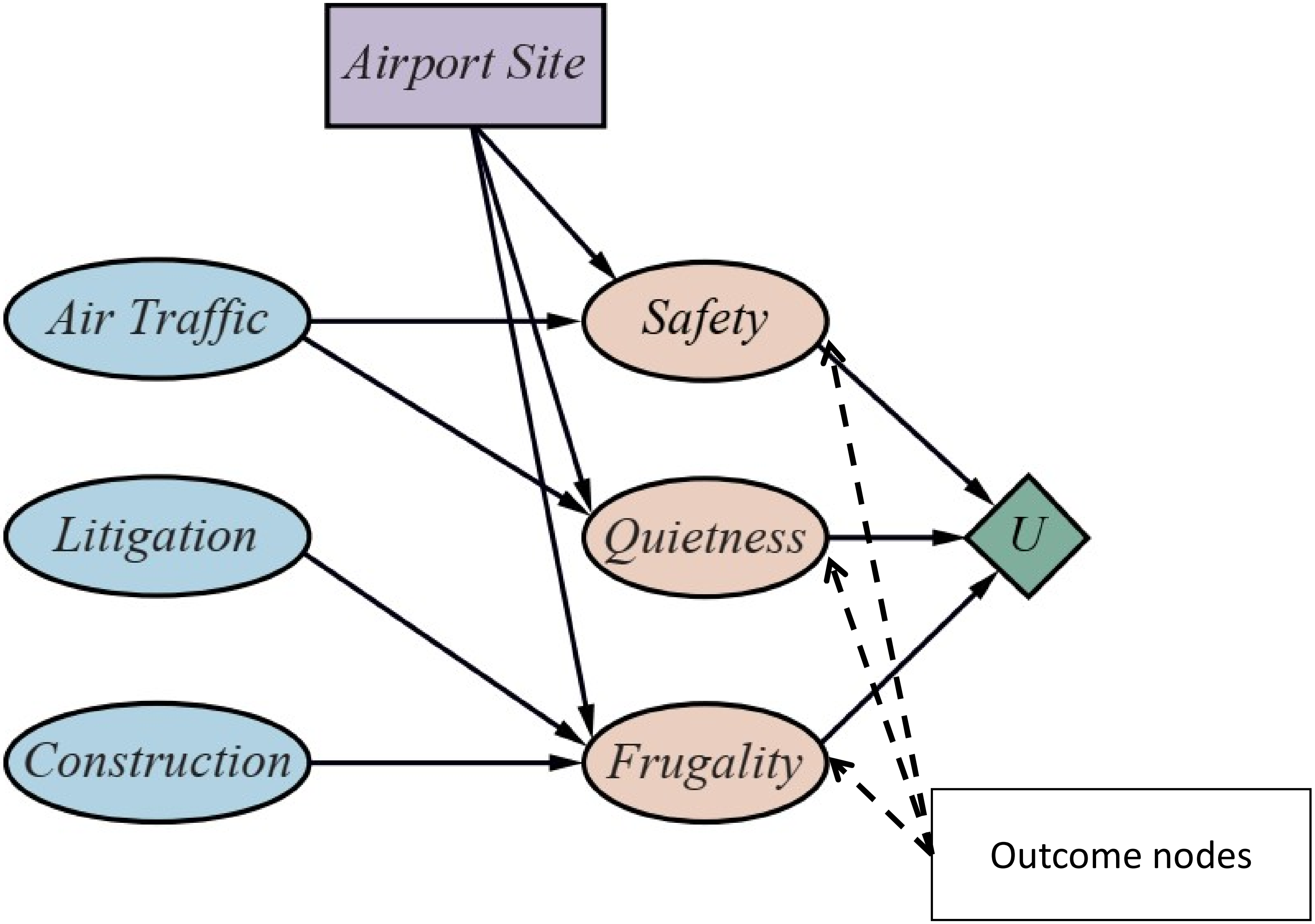
Airport Site decision changes *Safety*, *Quietness*, and *Frugality* nodes conditional distributions

Decision Network: Example

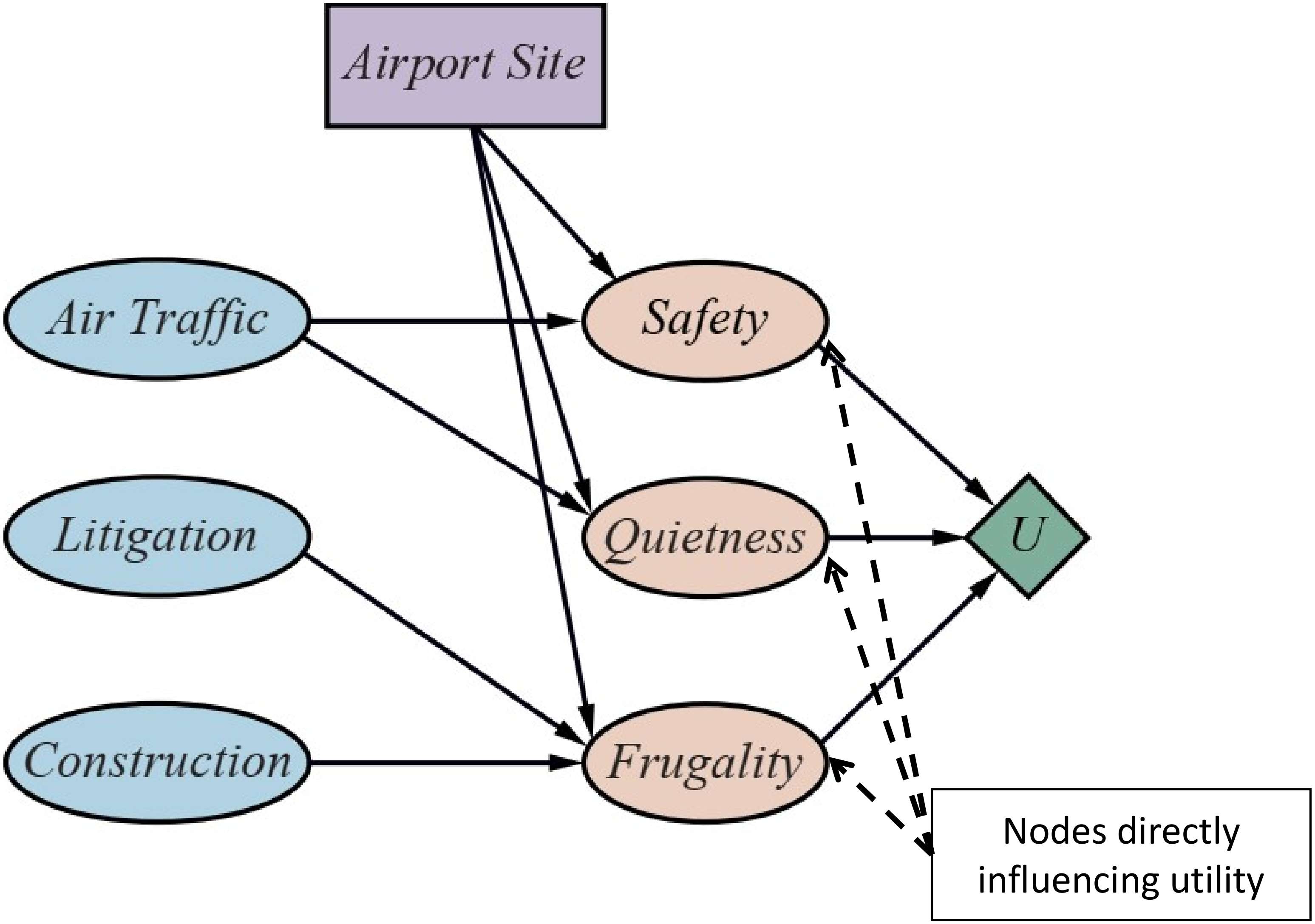


Parents of the utility node

Decision Network: Example



Decision Network: Example



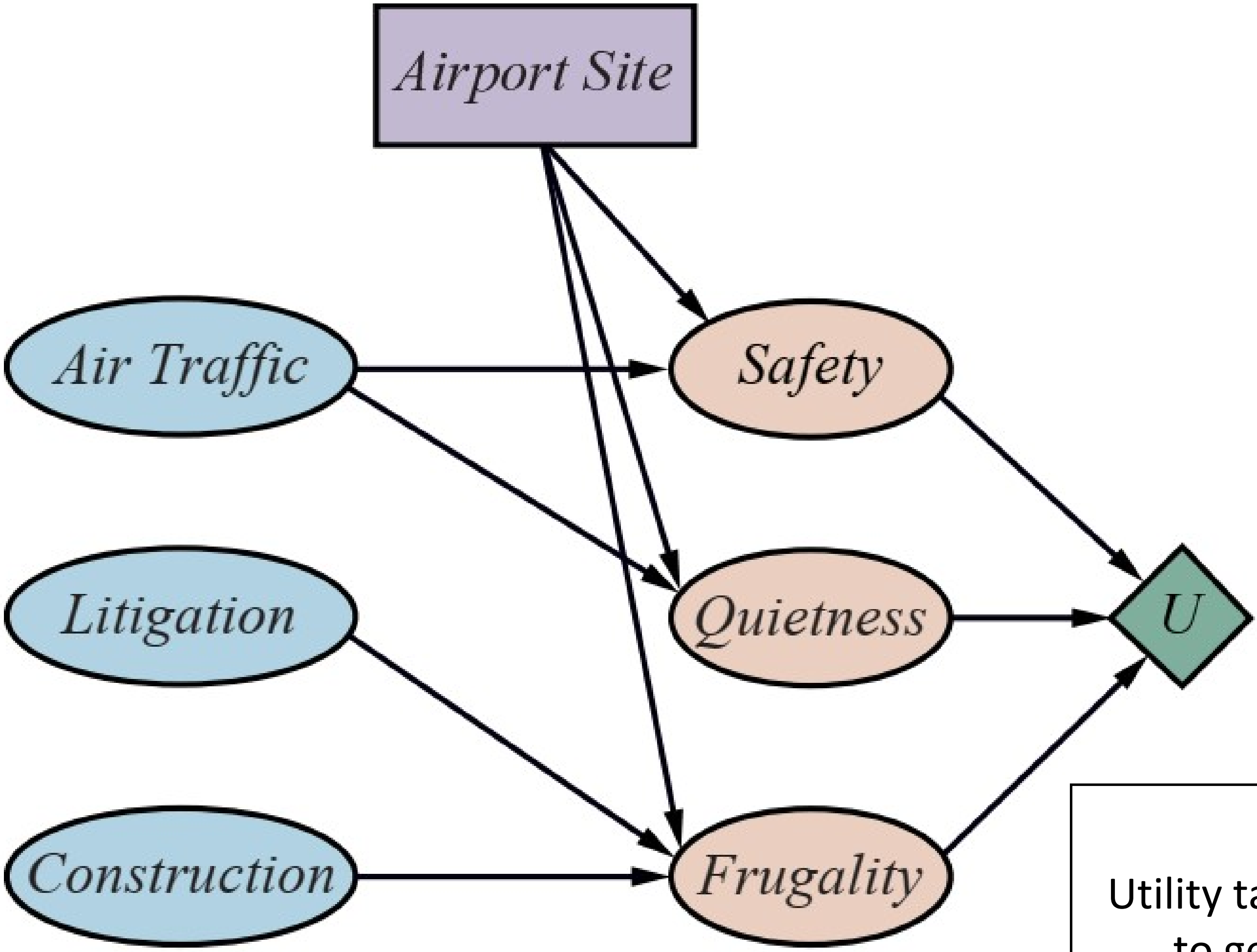
Nodes directly influencing utility

Decision Network: Evaluation

The algorithm for decision network evaluation is as follows:

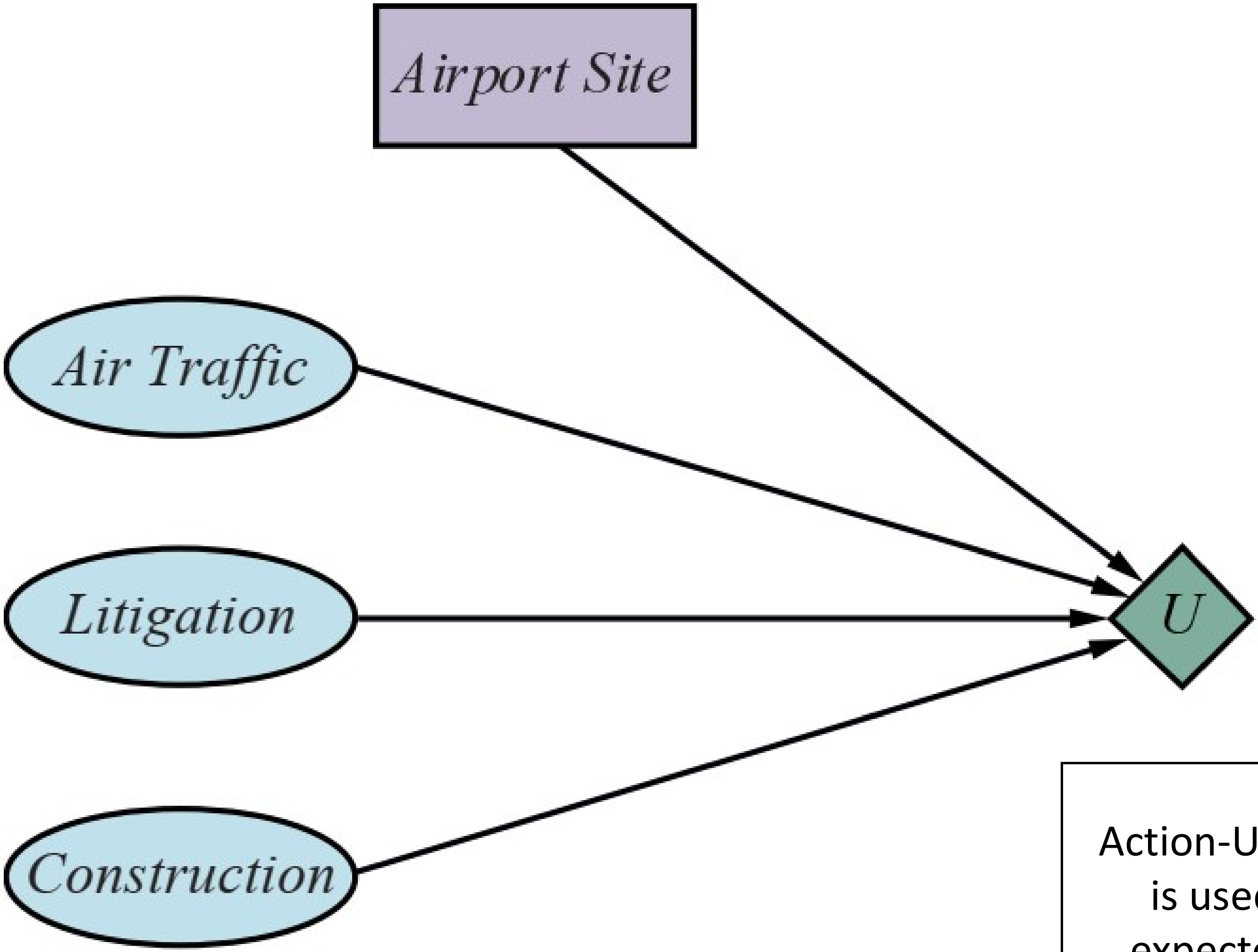
1. Set the evidence variables for the current state
2. For each possible value a of decision node:
   a. Set the decision node to that value
   b. Calculate the posterior probabilities for the parent nodes of the utility node
   c. Calculate the utility for the action / value a
3. Return the action with highest utility

Decision Network: Example



Utility table is used to get utility

Decision Network: Simplified Form



Action-Utility table is used to get expected utility