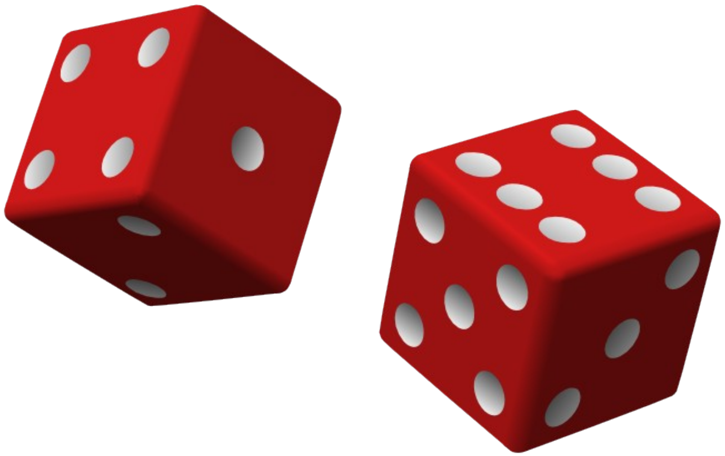
Artificial Intelligence

Lecture 1: Intro & Reasoning under uncertainty

R&N Chap. 1: 1.1
R&N Chap. 2: 2.1–2.3
R&N Chap. 13: all (recap)
R&N Chap. 14: 14.1, 14.2,14.4
optional: 14.3



IN4010 Artificial Intelligence Techniques

dr. F. Oliehoek, Prof. dr. C.M. Jonker, dr. Matthijs Spaan, Jinke He

Interactive Intelligence group
Department of Electrical Engineering, Mathematics and Computer Science

TU Delft
Delft University of Technology

Welcome!

- Teachers

Frans Oliehoek

Catholijn Jonker

Jinke He

Matthijs Spaan

- TAs:
  - Danyao Wang
  - Aayush Singh

TU Delft

Questions: stackoverflow

- Questions? We are using stackoverflow
  - https://stackoverflow.com/c/tud-cs/
  - (not brighspace forum)
  - use correct TAGs: “ait”

    (or “artificial-intelligence-techniques” , “in4010-12”)
    - https://stackoverflow.com/c/tud-cs/posts/tagged/405?tab=Newest

- How to use stackoverflow?
  - https://stackoverflow.com/c/tud-cs/articles/2347

Contact information: Brightspace
(Week 0: Course Information → Staff and Support)

TUDelft

Today

- Logistics…
- What is Artificial Intelligence (AI)?
  - And why do we care?
- Reasoning under uncertainty

Logistics

Course information…

- All information should be available at brighspace (under “contents → week 0: course information”)
  - course description, learning objectives, expected background, dates, etc.
  - let us know if you can’t find something

Schedule
(see brightspace for full schedule)

- Lectures Q1
  - on Monday, 13:45-15:45
  - Hybrid:
    - max 75 persons.
    - register in queue for a physical place
  - On campus: EWI Lecture hall Pi (building 36)
  - Online: Zoom Meeting
    - https://tudelft.zoom.us/j/99566627107?pwd=UzFrYzdmdTlpRjF0Rk1aRVNJREVXZz09
    - Meeting ID: 995 6662 7107
    - Passcode: 923772

Schedule
(see brightspace for full schedule)

- “Tutorial” Q1
  - on Fridays, 11:45-12:45
  - online – zoom:
    - https://tudelft.zoom.us/j/93791873795?pwd=MHlFVWlsbXBVSTNCQklneVQ3QWNrQT09
    - Meeting ID: 937 9187 3795
    - Passcode: 639078
  - This is **question hour**:
    - can ask questions about the material of **previous** week
  - first week: no tutorial.
- Q2… still in the making

Organizational (see Brightspace)

- Reading
  - Russell and Norvig (R&N) and additional material as indicated on Brightspace
  - Sections of R&N also indicated on slides

Tutorial Exercises

- Sheets with tutorial exercises will be made available
  - not graded
- The “tutorial” hour is meant to ask questions
  - It’s a “question hour”, not an additional lecture
  - Do the exercises in advance!

Practical assignment

- **Three** assignments
  - instructions will be published on Brightspace
- **implement** some AIT techniques
- **Group** work 4-5 people
- When? at your own time and place
- Graded. Together the practical assisgnments count for 20%.
  - no resit opportunity.

- Form groups of 4 to 5 people
  - right now & here,
  - or on forum

→ make sure you have ample programming experience in the team (both python and java)

→ register your team

Exam

- In Januari 2022
  - no mid-term
  - both multiple choice and open questions
  - counts for 80%

- Resit in Q3

Topics

- Approaching problems from
  - Single agent perspective
  - Multi-agent perspective
- We present a “decision-theoretic perspective”
  - probability for uncertainty
  - utility for goals
- Some keywords / techniques:
  - Bayesian networks and probabilistic inference
  - Rational decision making
  - Reinforcement learning and Markov Decision Problems
  - Game theory
  - Negotiation strategies
  - Adversarial and cooperative decision making

Related courses

Well... what not? E.g.:

- Algorithms for Intelligent Decision Making
- Intelligent Decision Making Project
- Machine Learning 1 & 2
- Deep learning
- Intelligent Decision Making Project
- Deep Reinforcement Learning
- Conversational Agents
- Information Retrieval
- Social Signal Processing
- Evolutionary Algorithms

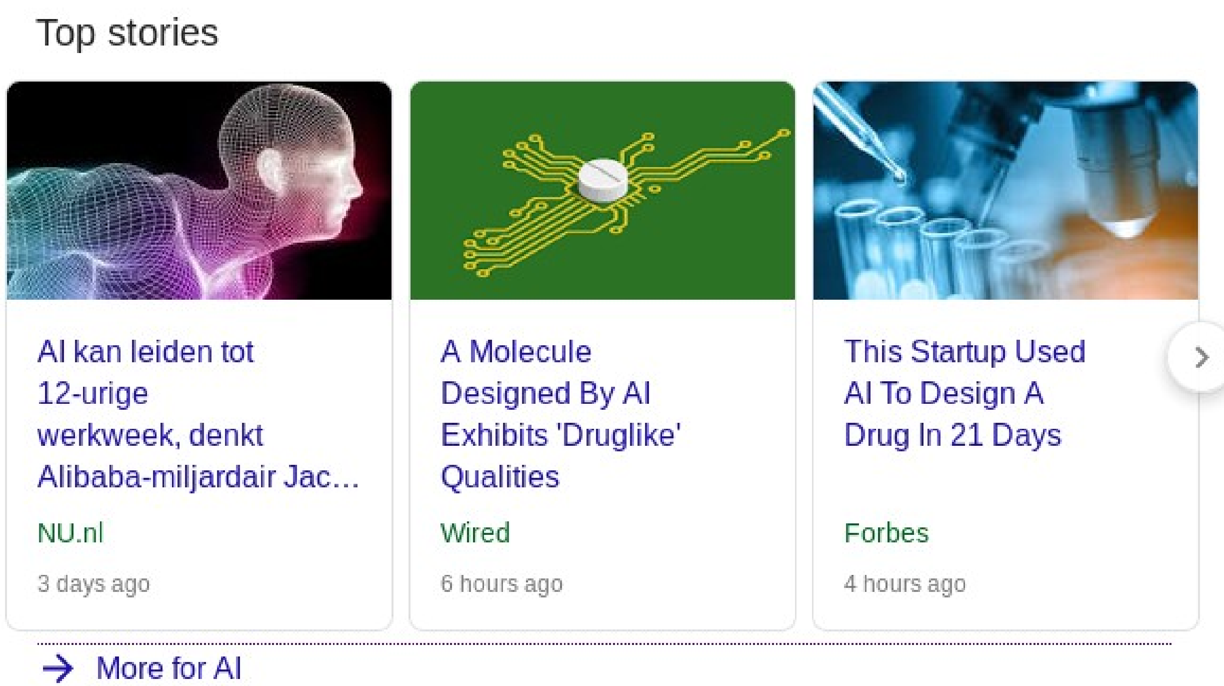
Why AI?

What is AI?

What is an intelligent agent?

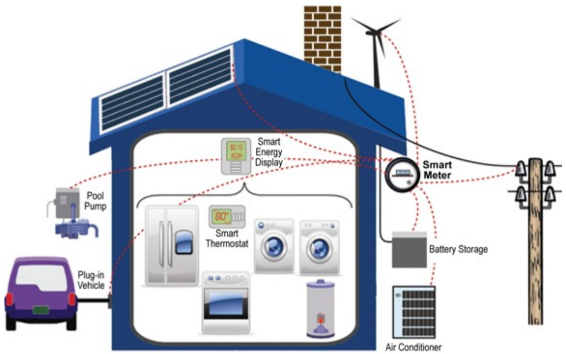
Why AI…?

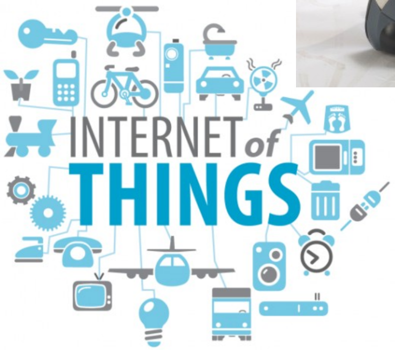
- Massive interest in AI…



Delegating Decision Making to Machines

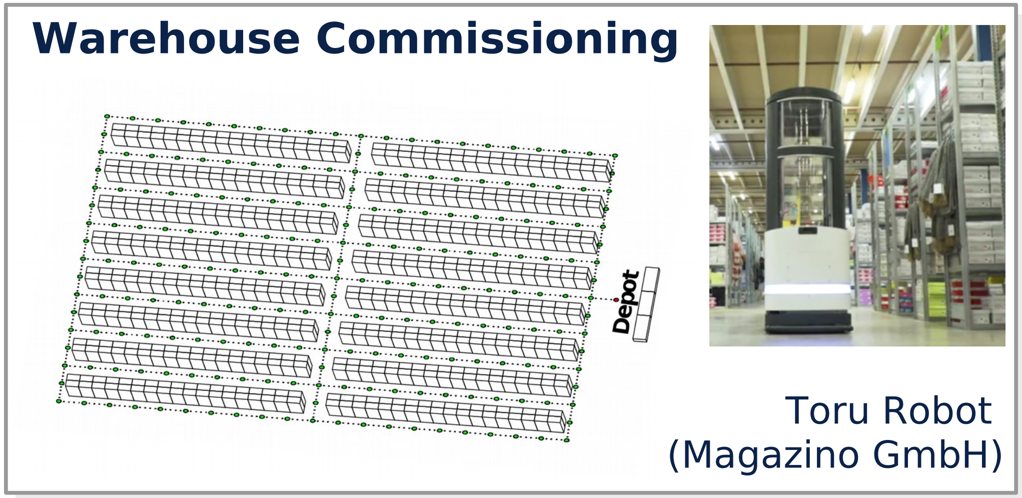
McKinsey: **Uptake of AI**: by 2025, machines will be able to learn, adjust, exercise judgment, & reprogram themselves



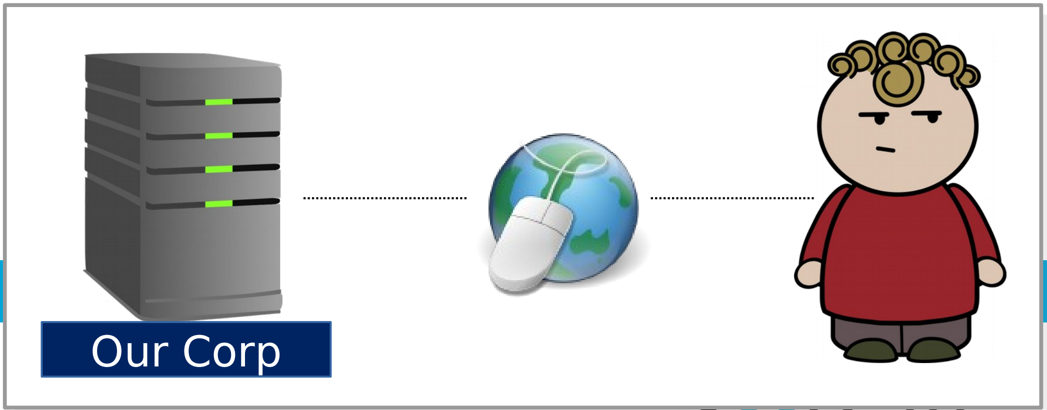


How can we...

- ...run an autonomous warehouse?



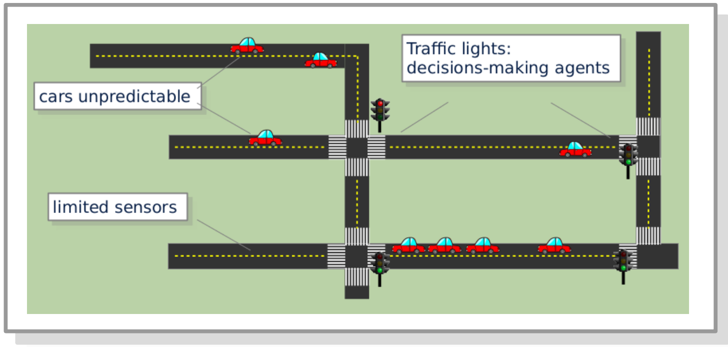
- ...interact with customers on-line?



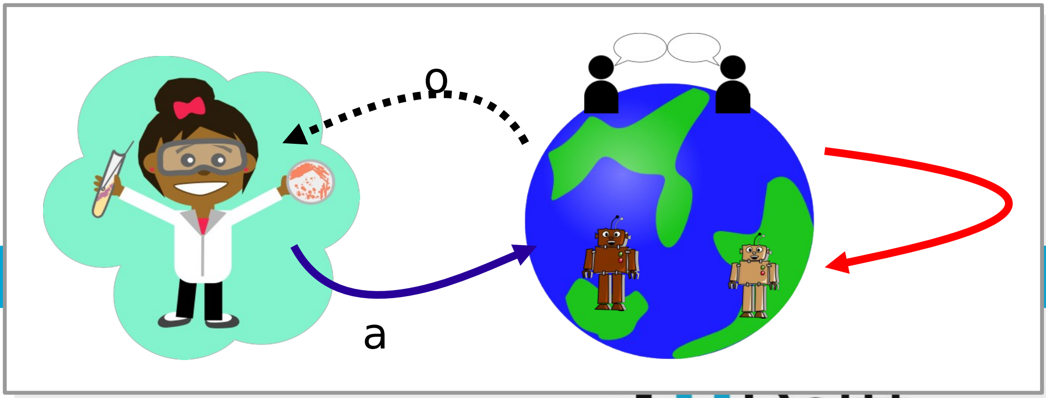
How can we…

- …beat Go Grandmasters?

- …coordinate traffic lights in a large city?



- …develop an artificial scientist?



What is Artificial Intelligence?

Many definitions have been proposed for Artificial Intelligence

Df 1: Machine ≥ human

- The science that tries to automate processes that humans so far do better than machines

Df 2: Understand and simulate

- The science that aims to understand natural intelligence so well that it can be simulated on a computer.

Df 3: Human and computer

- The science of developing intelligent software that supports humans.

What sort of agent system?

What kind of agent system is AI trying to engineer?

Biology
Psychology
Sociology
Cognitive Sc

Think Like Humans	**Think Rationally**
Act Like Humans	**Act Rationally**

Mathematics
Economy
Logic
Game Theory
Philosophy

Acting humanly: Turing test

- Turing (1950) "Computing machinery and intelligence"
  - "Can machines think?"
- Operational test for intelligent behavior: Imitation Game
- Suggested major components of AI: knowledge, reasoning, language understanding, learning.

- Pro: clear yes/no answer
- Cons:
  - Not reproducible, constructive, or amenable to mathematical analysis.
  - It does not provide any guidelines for building agents!

Thinking humanly: Cognitive science

- What is thinking humanly?
  - Requires scientific theories of internal activities of the brain...
  - Two scientific approaches:
    - Predicting and testing behavior of human subjects (top-down)
    - Direct identification from neurological data (bottom-up)
- Both approaches (roughly cognitive science and cognitive neuroscience) are now distinct from AI

- Pro: CogSci provides potentially useful theories for AI engineers!
- Con:
  - human specifics such as memory limitations which are(?) irrelevant for machine intelligence

Thinking rationally: Laws of Thought

- Goes back to Aristotle: what are correct arguments / thoughts?
  - Socrates is a man; all men are mortal; therefore …
- Use logic to derive the ‘right’ conclusions
- Pros:
  - Theoretically well-founded.
- Cons:
  - Difficult to capture the worlds in crisp logical statements
  - Perhaps not all intelligent behavior is mediated by logical deliberation.

Acting rationally: the rational agent

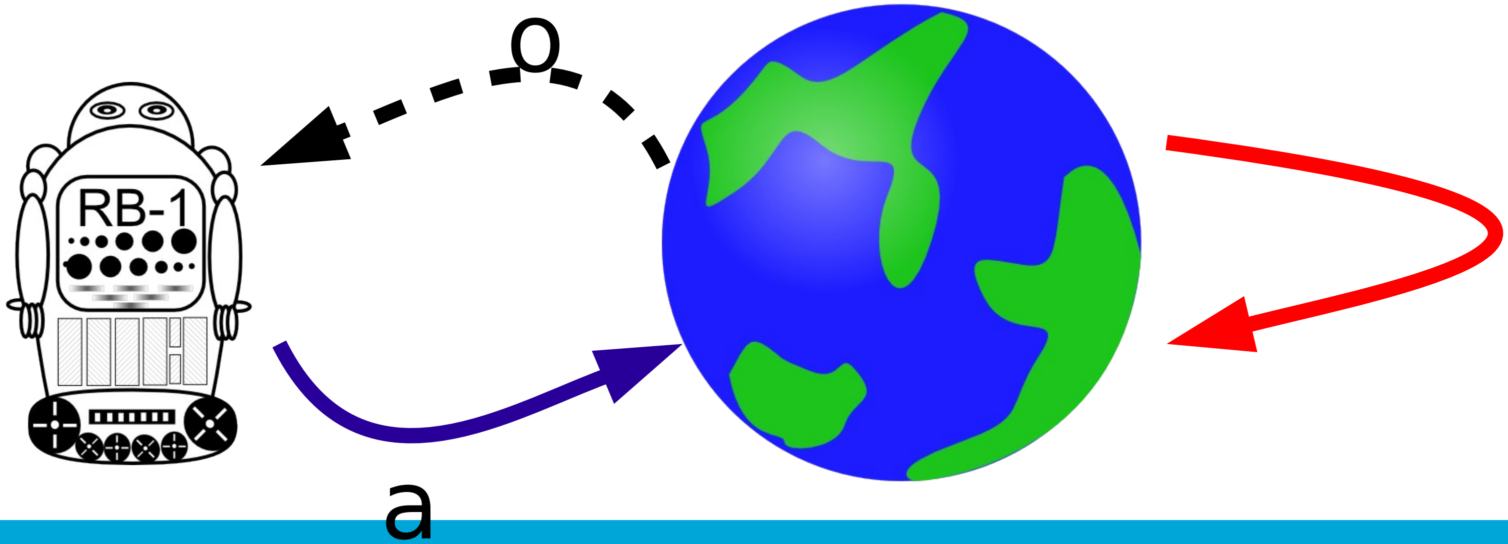
- Rational behavior: doing “the right thing”…
  - expected to **maximize goal achievement** or a different notion of **performance**, given the available information.

- Pros:
  - Also theoretically well-founded
  - More general than “law of thought”
    (correct reasoning is just one way of getting correct actions)
  - mostly quantitative: well-suited for scientific approach

- Cons:
  - Quantitative… → can be hard to obtain the “numbers”, or to relate the theory to **human** performance.

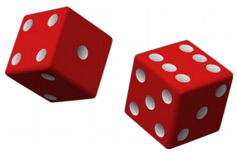
Definition of Agent
According to Russel&Norvig (ch.2)

*"An **agent** is anything that can be viewed as perceiving its **environment** through **sensors** and acting upon that environment through **actuators**"*

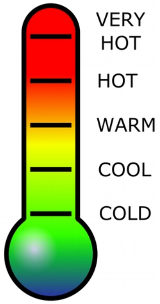


What is uncertainty and why AI needs to care?

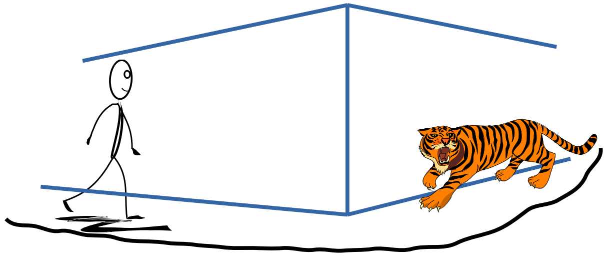
Why care about uncertainty?



- Stochasticity:
  - outcome of actions uncertain
  - environment can fluctuate

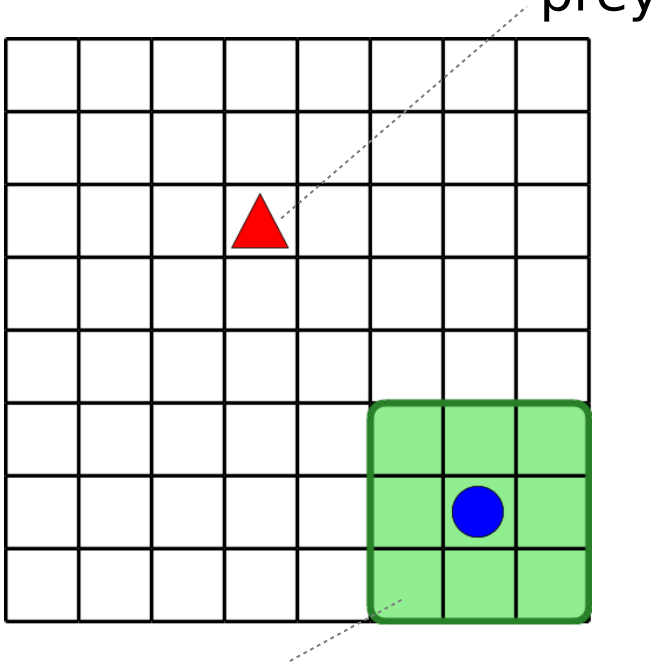


- State uncertainty:
  - sensor noise
  - limited sensor

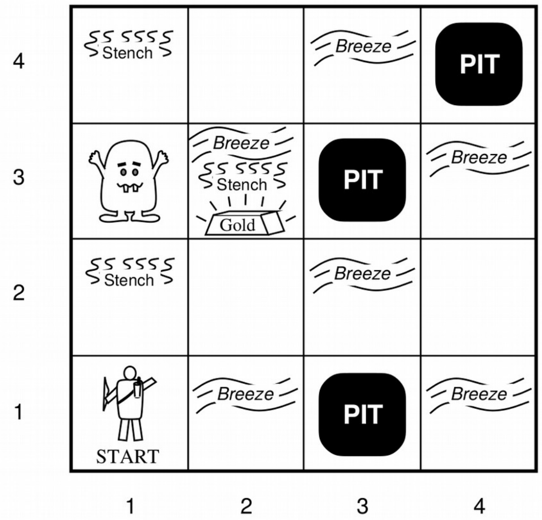


Logic for dealing with uncertainty
(E.g., R&N chap 7)

- Represent world using facts over which we can reason
  - e.g.: predator-prey
  - state=(-3,4)
- Maintain a **belief**:
  - set of possible states!
  - b={(-6,-1),(-6,0),(-6,1)…}



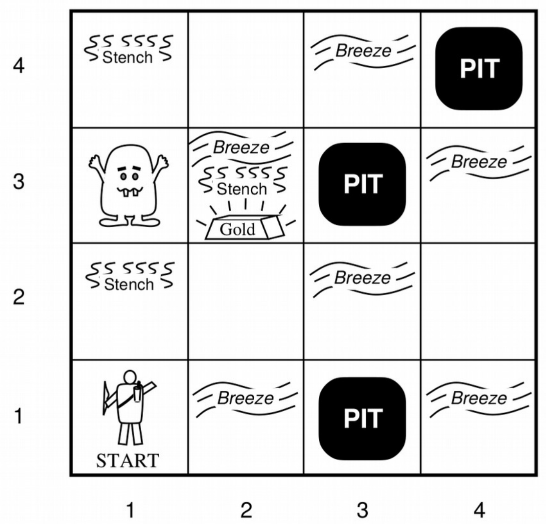
Scaling this…



- Works for simple problems
  - static prey
  - simple predictable movements
  - "Wumpus world" (R&N, sect. 7.2)

- Planning to get to the airport in to make our flight…
  - a_t = "leave in t minutes"
  - logical conclusion:
    a_{25} will get me on time if
    - there is no traffic jam, and
    - the car does not break down, and
    - do not need to get gas, and
    - I don't get in an accident, and
    - …

Scaling this…



- Works for simple problems
  - static prey
  - simple predictable movements
  - "Wumpus world" (R&N, sect. 7.2)

- Planning to get to the airport in to make our flight…
  - a_t = "leave in t minutes"
  - logical conclusion:
    a_{25} will get me on time if
    - there is no traffic jam, and
    - the car does not break down, and
    - do not need to get gas, and
    - I don't get in an accident, and
    - …

Too many things can go wrong…!

"qualification problem"
need to specify all exceptions of desired effects of actions.

TU Delft

Different approach…

- Not all these possible outcomes are likely…
- Base decision on **degree of belief** *b(o|a)*
  - in outcomes *o* – *{on_time, too_late}*
  - given alternative actions *a* – $\{a_{10}, a_{25}, a_{40}, \ldots\}$
- Select action with highest `believed' **utility**:

$$U(a) = \sum_o u(o) * b(o|a)$$

Different approach…

- Not all these possible outcomes are likely…
- Base decision on **degree of belief** $b(o|a)$
  - in outcomes o – *{on_time, too_late}*
  - given alternative actions a – $\{a_{10}, a_{25}, a_{40}, \ldots\}$
- Select action with highest `believed’ **utility**:

$$U(a) = \sum_o u(o) * b(o|a)$$

- how does this help?
- how do we form these beliefs?

TUDelft

Degrees of belief

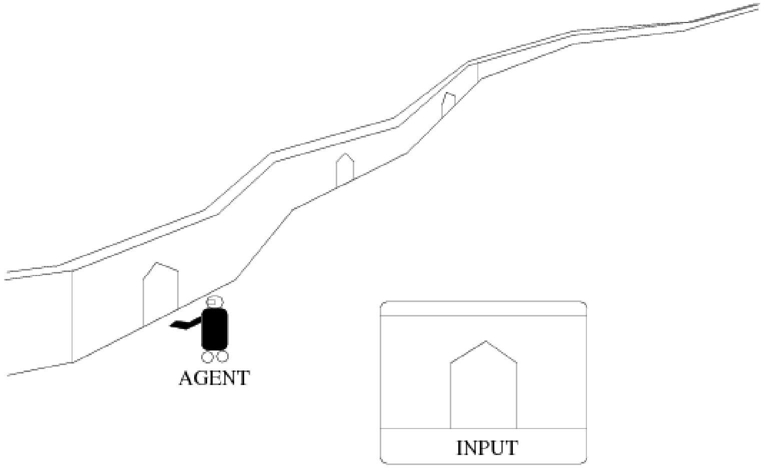
- Many people advocate: probability
- Leads to **maximum expected utility**

$$U(a) = \sum_o u(o) * P(o|a)$$

- also “decision theory” (=probability + utility)

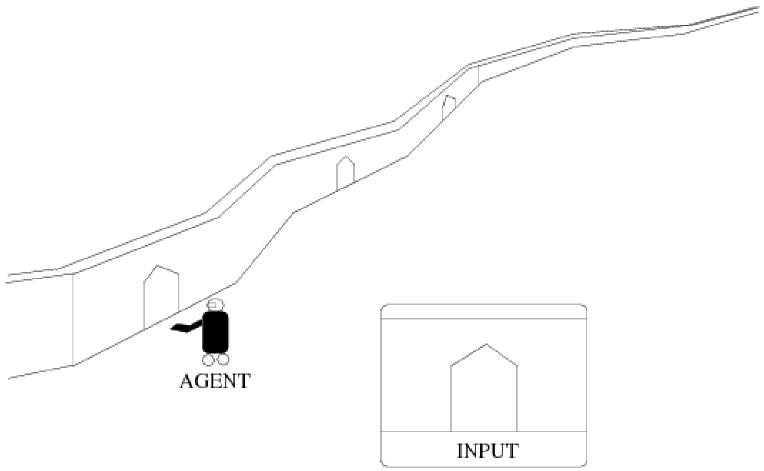
State Uncertainty

- Traffic example: **outcome uncertainty**
- How about: **state uncertainty?**



State Uncertainty

- Traffic example: **outcome uncertainty**
- How about: **state uncertainty?**



Similar…!

- now state *s* is hidden
- utility of action depends on *s*

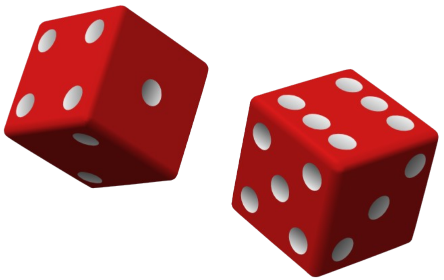
$$U(a) = \sum_s u(s,a) * b(s)$$

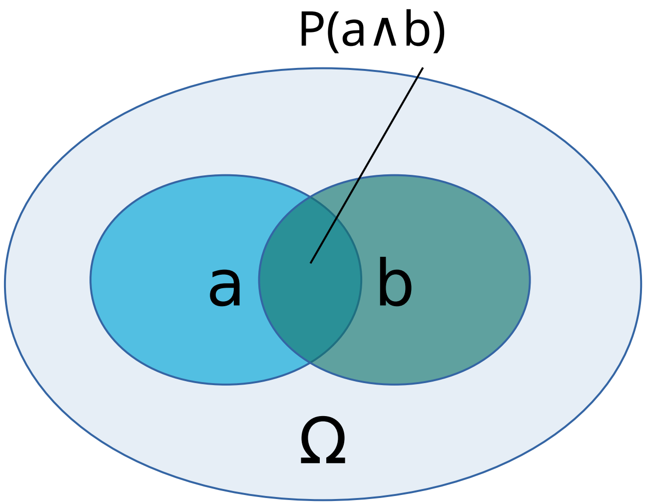
TU Delft

What you should know about probability...

Short answer:
All of Russel&Norvig(v3) Chap 13

- events, random variables
- joint probability
- inference by enumeration
- independence
- conditional independence
- Bayes Rule





Again: why probability?

De Finetti's argument

1) (Non-negativity) $P(A) \geq 0$, for all $A \in F$.
2) (Normalization) $P(\Omega) = 1$.
3) (Finite additivity) $P(A \vee B) = P(A) + P(B)$ for all $A, B \in F$ such that $A \cap B = \emptyset$.

Bruno de Finetti:

If agent's beliefs violate the axioms of probability, then there exists a combination of bets against it which it is willing to accept that guarantees it will lose money, every time.

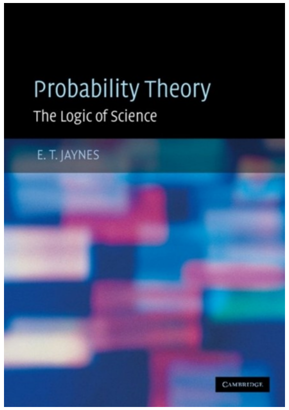
Example

u(6:4 against a) = -6 * bel(a) + 4 * bel(¬a)
= -6 * 0.4 + 4 * 0.6 = 0

A Dutch book:

prop.	belief	taken bet	a,b	a,¬b	¬a,b	¬a,¬b
a	0.4	6:4 against a	-6	-6	4	4
b	0.3	7:3 against b	-7	3	-7	3
a∨b	0.8	8:2 on a∨b	2	2	2	-8
			-11	**-1**	**-1**	**-1**

Cox' Theorem
(See Jaynes 2003)



- Desiderata:
  - Degrees of plausibility: represented by **real numbers**
  - **Qualitative** correspondence how humans reason
  - **Consistency**: If a conclusion can reached in more ways, then every possible way must lead to the same result

→ Need to use probability to represent plausibility

- "Probability theory is nothing but common sense reduced to calculation." — Laplace, 1819

TU Delft

Bayes rule

“One rule to rule them all”

If you are going to remember just one thing…

- …remember Bayes' rule:

  $$P(A|B) = P(B|A)P(A) \,/\, P(B)$$

directly from product rule:
$\Leftrightarrow$ $P(A|B)\,P(B) = P(B|A)P(A)$
$\Leftrightarrow$ $P(A,B) = P(B,A)$

Generalized form given background evidence ***e***:

$$P(A|B,\mathbf{e}) = P(B|A,\mathbf{e})P(A|\mathbf{e}) \,/\, P(B|\mathbf{e})$$

- Important why..?

  (What is its importance for an intelligent agent?)

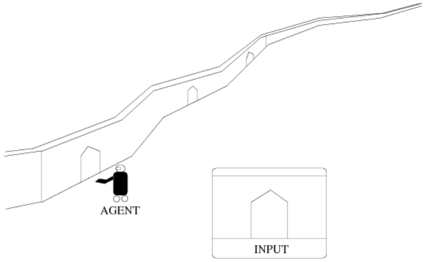
TU Delft

It allows to update a belief

- Rewrite with State and a particular observation *o*:

  $\boldsymbol{P}(State|o) = \boldsymbol{P}(o|State)\boldsymbol{P}(State) / P(o)$

  - ***P****(State)* is our prior belief
  - so we can update our belief, based on observations!



Many observations…

- How to deal with **many observations**?

  $\mathbf{P}(State|o_1,o_2,o_3,\ldots)=\alpha\,\mathbf{P}(o_1,o_2,o_3,\ldots|State)\mathbf{P}(State)$

  - we don't know the observation sequence in advance…
  - representing $\mathbf{P}(o_1,o_2,o_3,\ldots|State)$ with a table does not scale…

Many observations...

- How to deal with **many observations**?

  $\mathbf{P}(State|o_1,o_2,o_3,\ldots)=\alpha\, \mathbf{P}(o_1,o_2,o_3,\ldots|State)\mathbf{P}(State)$

  - we don't know the observation sequence in advance...
  - representing $\mathbf{P}(o_1,o_2,o_3,\ldots|State)$ with a table does not scale...

- Solution:
  - conditional independence can help significantly

It allows to *maintain* a belief

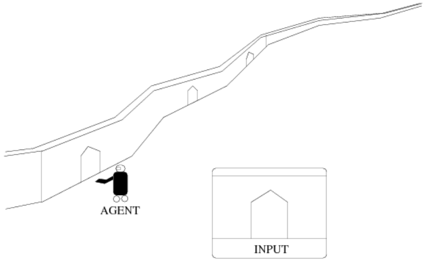
- Given conditional independent observations
  $\boldsymbol{P}(o_1,o_2|State) = \boldsymbol{P}(o_1|State)\boldsymbol{P}(o_2|State)$

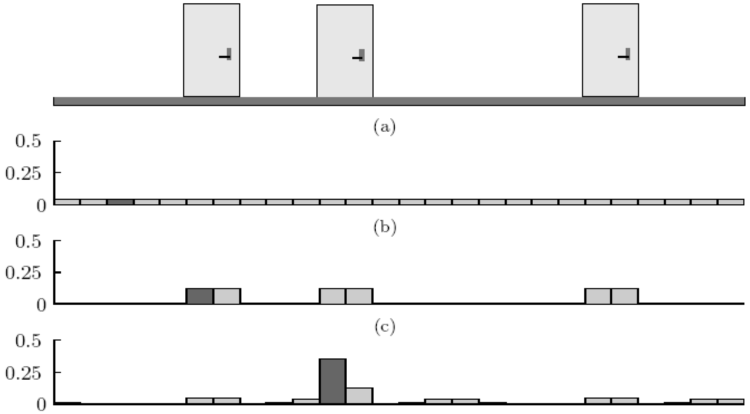
- ...we can also sequentially update:

  $\boldsymbol{P'}(State) := \boldsymbol{P}(o_1|State)\boldsymbol{P}(State) / P(o_1)$
  $\boldsymbol{P''}(State) := \boldsymbol{P}(o_2|State)\boldsymbol{P'}(State) / P(o_2)$

  - then $\boldsymbol{P''}(State)=\boldsymbol{P}(o_1,o_2|State)$
  - (Exercise!)

- We will see later how
  to incorporate robot movement over time





Compactly representing probability distributions:

Bayesian Networks

Compact Representations

- Recap:
  - Probability for beliefs – strong arguments
  - Bayes rule: allows updating of beliefs
  - The challenge: scaling to many variables
    - **joint probability tables don’t scale**

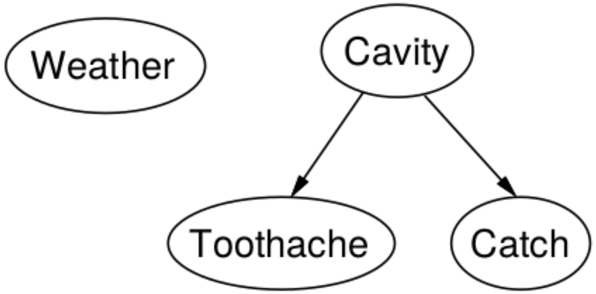
  - Important ‘hammers’:
    - independence – rare
    - conditional independence (CI) – much more common

- **Bayesian networks** use CI to represent complex problems…

Bayesian Networks (BNs)

- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ "directly influences")
  - a conditional distribution for each node given its parents:
    $\mathbf{P}(X_i|Parents(X_i))$

Topology of network encodes conditional independence assertions:



$Weather$ is independent of the other variables

$Toothache$ and $Catch$ are conditionally independent given $Cavity$

The Burglar Network

Example

I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometimes it's set off by minor earthquakes. Is there a burglar?

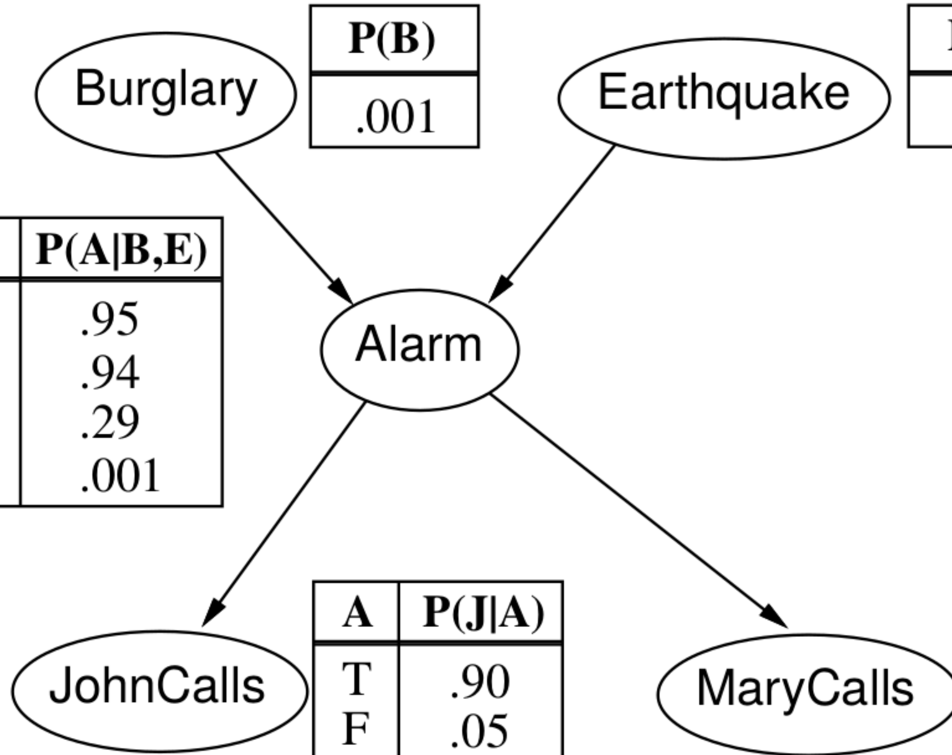
Variables: $Burglar$, $Earthquake$, $Alarm$, $JohnCalls$, $MaryCalls$
Network topology reflects "causal" knowledge:

- A burglar can set the alarm off
- An earthquake can set the alarm off
- The alarm can cause Mary to call
- The alarm can cause John to call

TU Delft

The Burglar Network

Example contd.



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

This represents P(B,E,A,J,M)

- but compactly using small **conditional prob. tables (CPTs)**

… how many parameters?

BNs in Practice

- huge number of applications...
  - disaster victim identification
  - Petrophysical decision support (oil, gas drilling)
  - process analysis
  - etc.

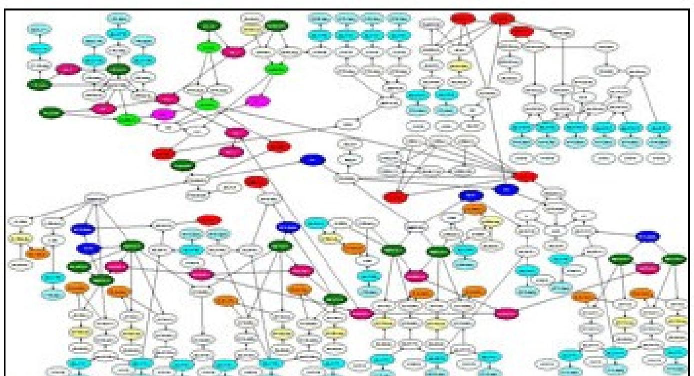
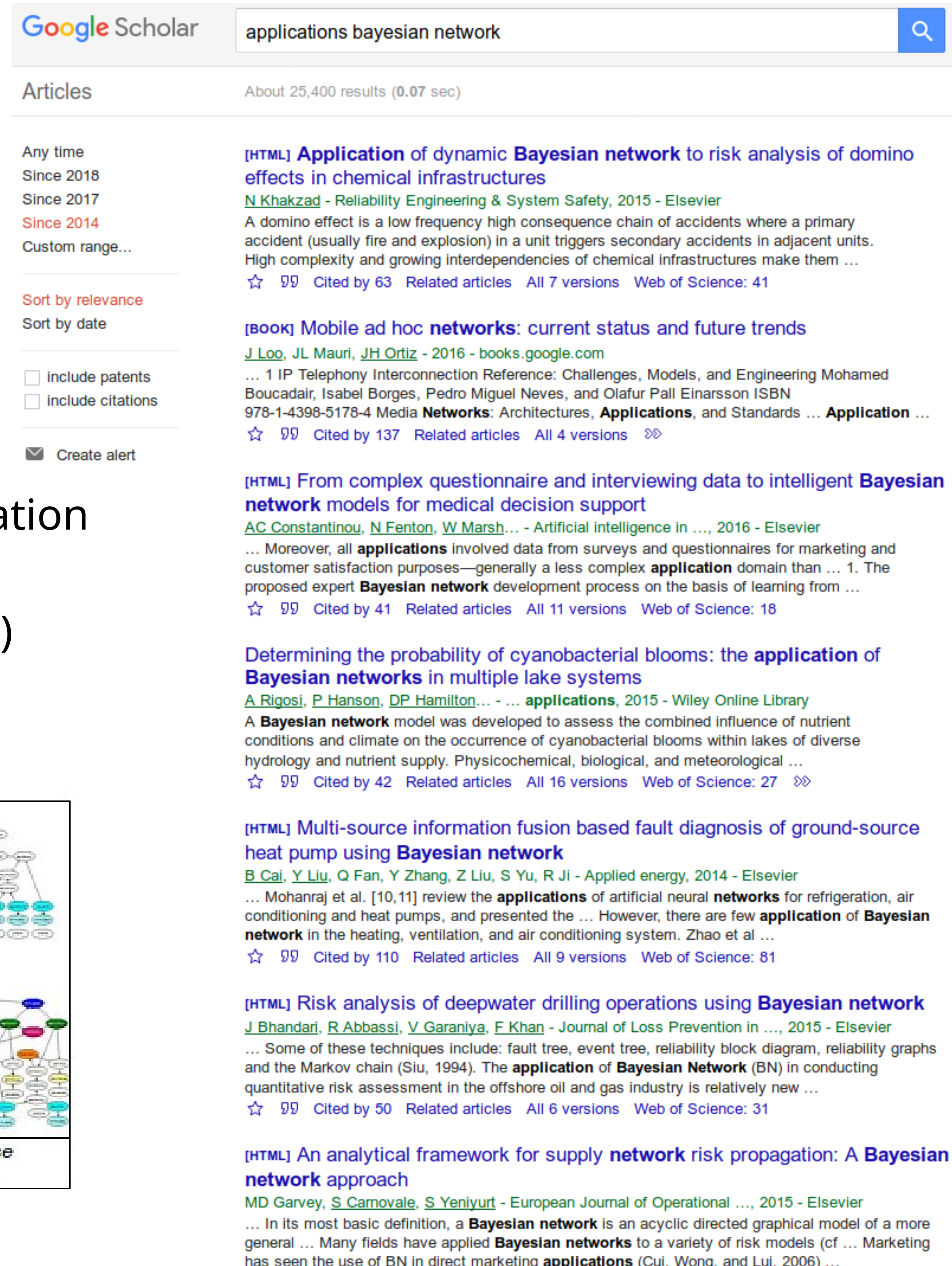


Figure 2: An example of a Bayesian Network for root cause analysis of process operation.



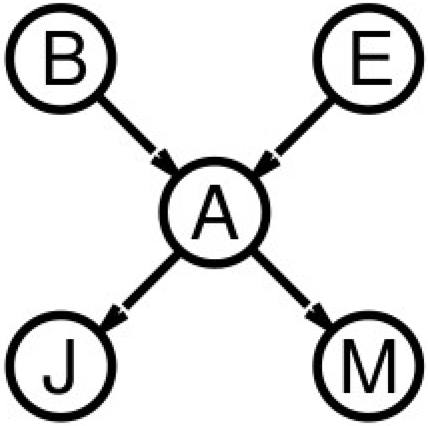
Semantics: global & local

"Global" semantics defines the full joint distribution as the product of the local conditional distributions:

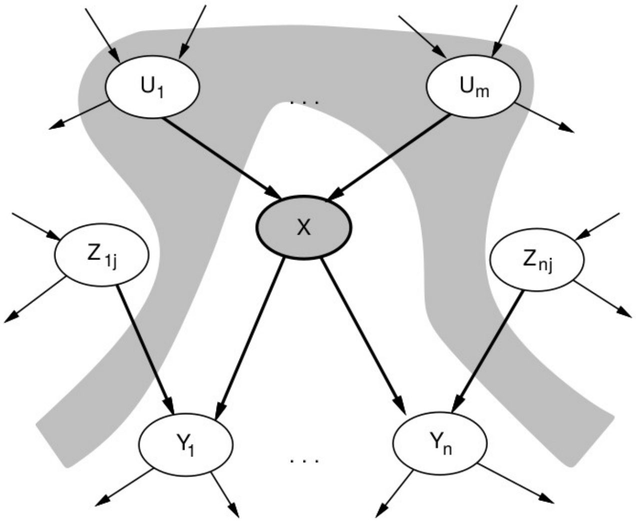
$$P(x_1, \ldots, x_n) = \Pi_{i=1}^{n} P(x_i | parents(X_i))$$

e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$$\begin{aligned} &= P(j|a)P(m|a)P(a|\neg b, \neg e)P(\neg b)P(\neg e) \\ &= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 \\ &\approx 0.00063 \end{aligned}$$



Local semantics: each node is conditionally independent of its nondescendants given its parents



Semantics: global & local

"Global" semantics defines the full joint distribution as the product of the local conditional distributions:

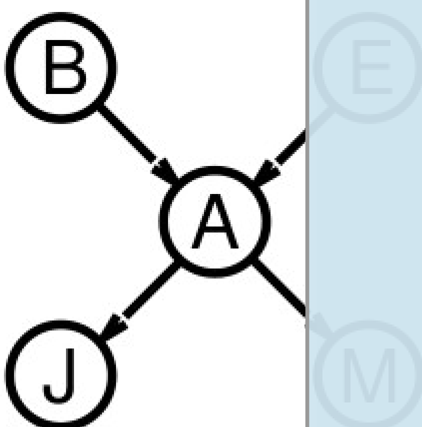
$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$$

e.g., $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

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

$= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998$

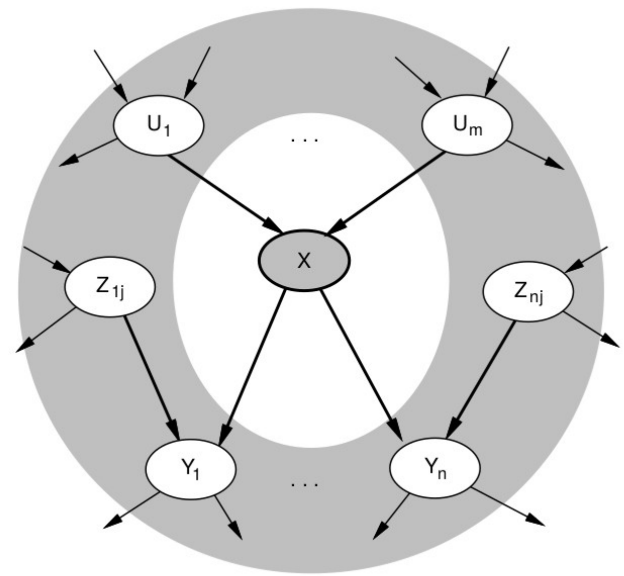
≈ 0.00063



Local semantics: each node is conditionally independent of its nondescendants given its parents

Local semantics, a bit stronger:

Each node is conditionally independent of all others given its Markov blanket: parents + children + children's parents



Constructing BNs

- How do we decide how to draw the arrows…?
  - Each PD P(x) might be representable by many BNs…?

- 1 firm rule:
  all conditional independencies implied by the BN need to hold in P(x)

- But many of these may have
  - unnecessary arrows
  - more parameters needed

- Rule of thumb: **put arrows in causal direction**
  (so from cause to effect)

- More details: see R&N!

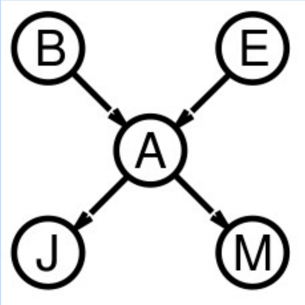
More considerations
(R&N: 14.3 - optional)

- Compact representations for the conditional probabilities
- Dealing with continuous variables
  - the graphical structure can stay the same!

(Exact) Inference

Inference Tasks

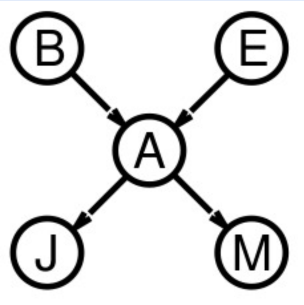
P(B|j,m)?



- A typical query:
  - What is ***P****(X|**E**=**e**)* ?
  - X – query variable
  - **e** – the values of observed evidence vars **E**
  - **Y** – any other variables that are not observed ("**hidden variables**")

Inference Tasks

P(B|j,m)?



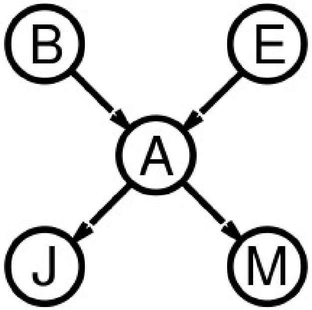
- A typical query:
  - What is ***P**(X|**E=e**)* ?
  - X – query variable
  - **e** – the values of observed evidence vars **E**
  - **Y** – any other variables that are not observed ("**hidden variables**")

- 'conditional probability query'
- others:
  - P(a) – marginal prob. query
  - max_x P(x) – max. a posteriori (MAP) query

The Burglar Network again

Simple query on the burglary network:

$\mathbf{P}(B|j,m)$
$= \mathbf{P}(B,j,m)/P(j,m)$
$= \alpha\mathbf{P}(B,j,m)$
$= \alpha \sum_e \sum_a \mathbf{P}(B,e,a,j,m)$



but, we want to avoid constructing **P**(B,E,A,J,M)...

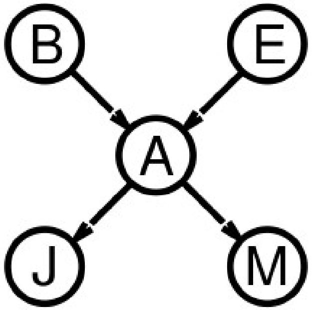
Rewrite full joint entries using product of CPT entries:

$\mathbf{P}(B|j,m)$
$= \alpha \sum_e \sum_a \mathbf{P}(B)P(e)\mathbf{P}(a|B,e)P(j|a)P(m|a)$
$= \alpha\mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e)P(j|a)P(m|a)$

The Burglar Network again

Simple query on the burglary network:

$$\mathbf{P}(B|j,m)$$
$$= \mathbf{P}(B,j,m)/P(j,m)$$
$$= \alpha \mathbf{P}(B,j,m)$$
$$= \alpha \textstyle\sum_e \sum_a \mathbf{P}(B,e,a,j,m)$$



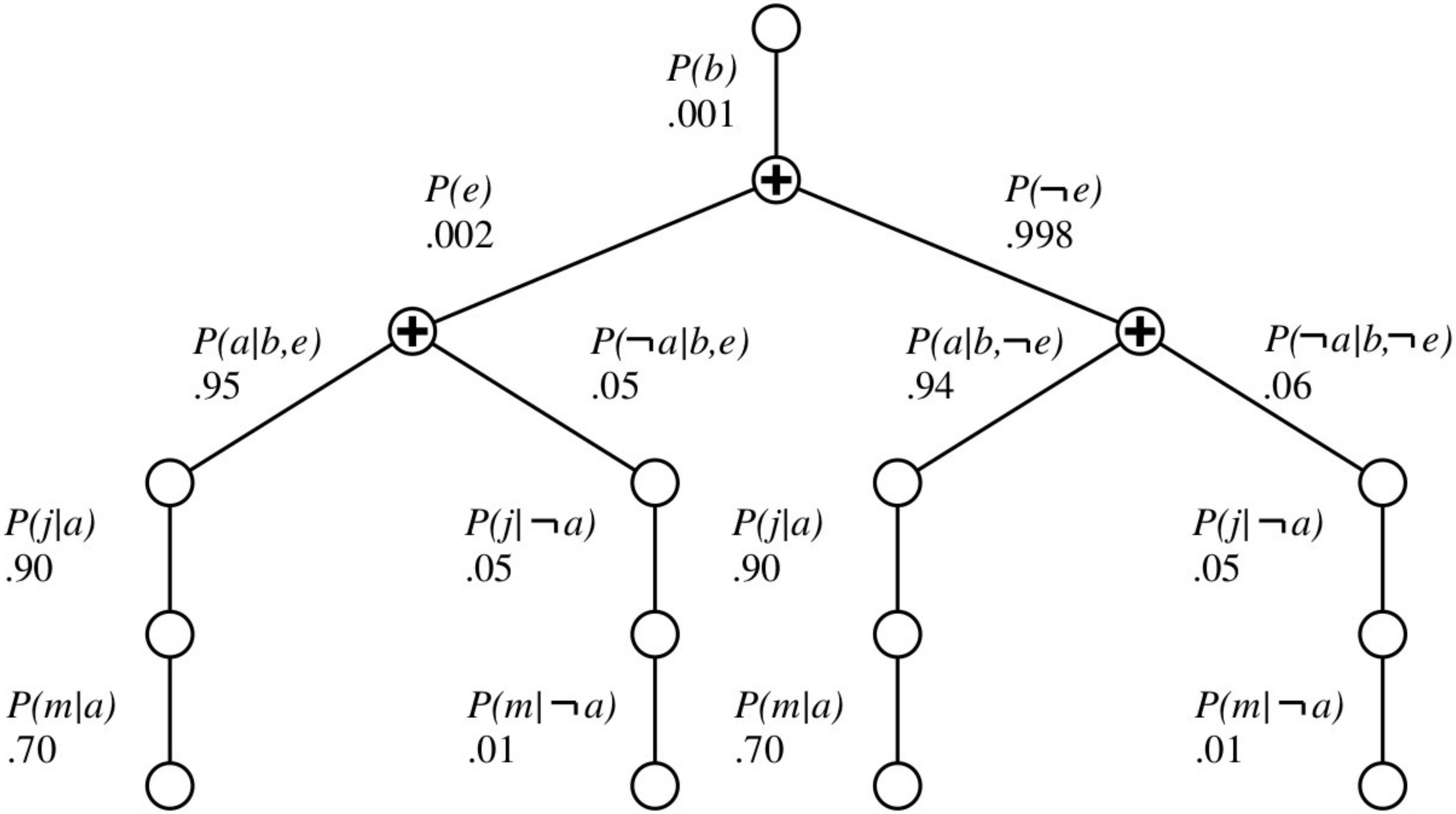
but, we want to avoid constructing **P**(B,E,A,J,M)…

- loop through variable values, to compute summations
- compute *P(b,e,a,j,m)* "on-the-fly"
- called inference by **enumeration** or **search**

Rewrite full joint entries using product of CPT entries:

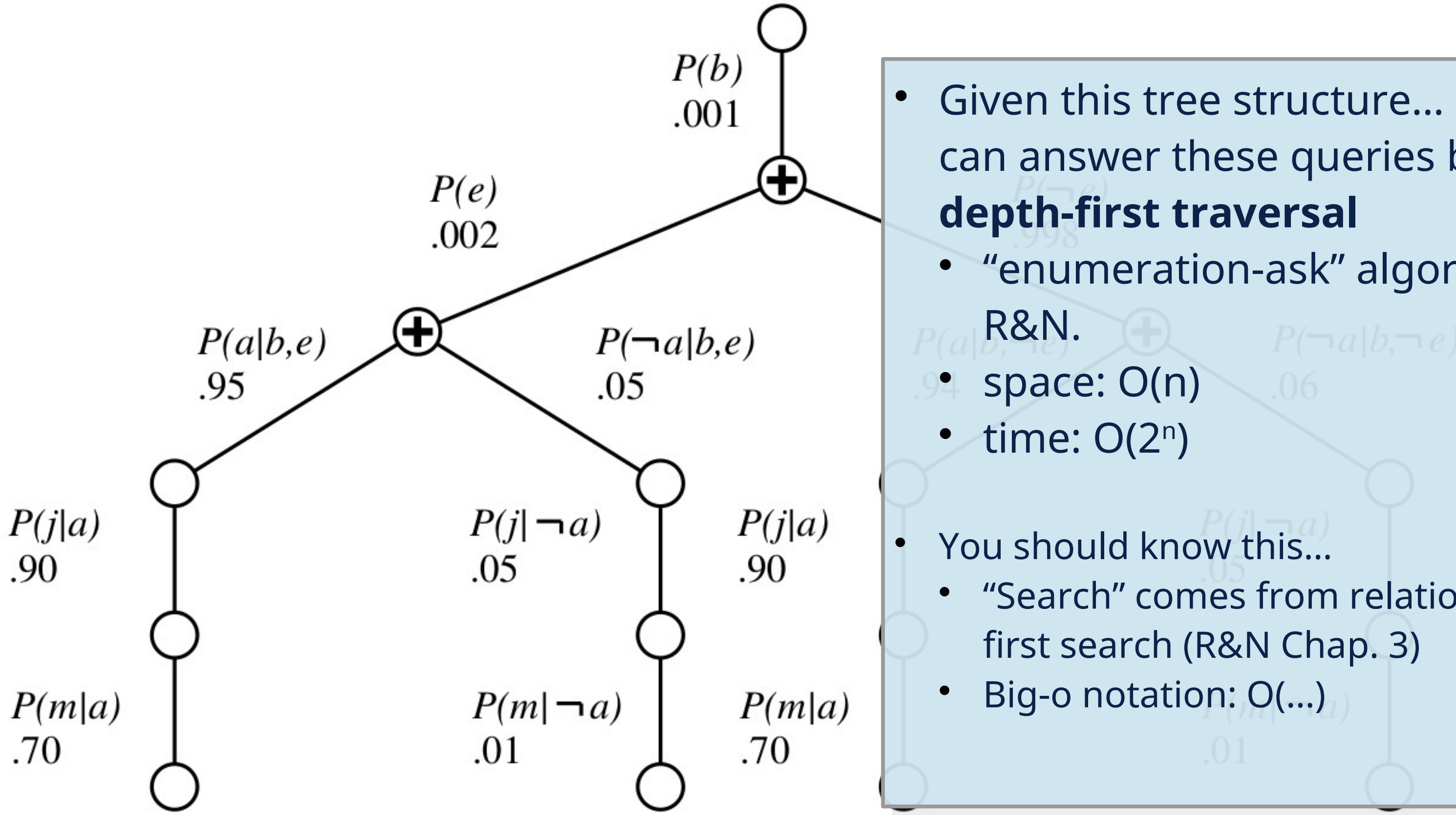
$$\mathbf{P}(B|j,m)$$
$$= \alpha \textstyle\sum_e \sum_a \mathbf{P}(B)P(e)\mathbf{P}(a|B,e)P(j|a)P(m|a)$$
$$= \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \sum_a \mathbf{P}(a|B,e)P(j|a)P(m|a)$$

Enumeration Tree



$$\mathbf{P}(B|j,m) = \alpha \mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e) P(j|a) P(m|a)$$

Enumeration Tree



- Given this tree structure… can answer these queries by **depth-first traversal**
  - "enumeration-ask" algorithm in R&N.
  - space: $O(n)$
  - time: $O(2^n)$

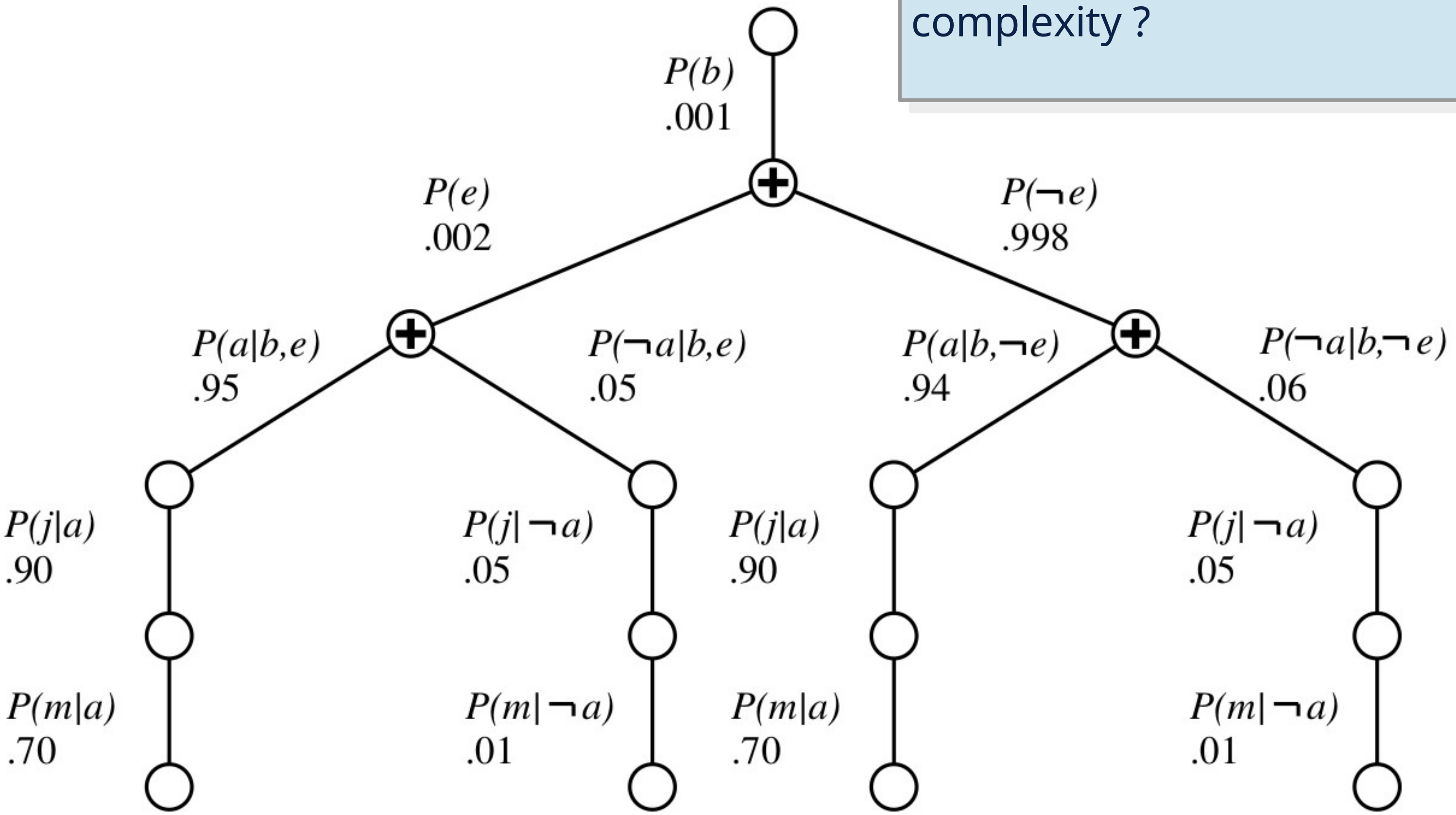
- You should know this…
  - "Search" comes from relation to depth-first search (R&N Chap. 3)
  - Big-o notation: O(…)

$$\mathbf{P}(B|j,m) = \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \sum_a \mathbf{P}(a|B,e) P(j|a) P(m|a)$$

TU Delft

Enumeration Tree

Can we improve upon $O(2^n)$ time complexity ?

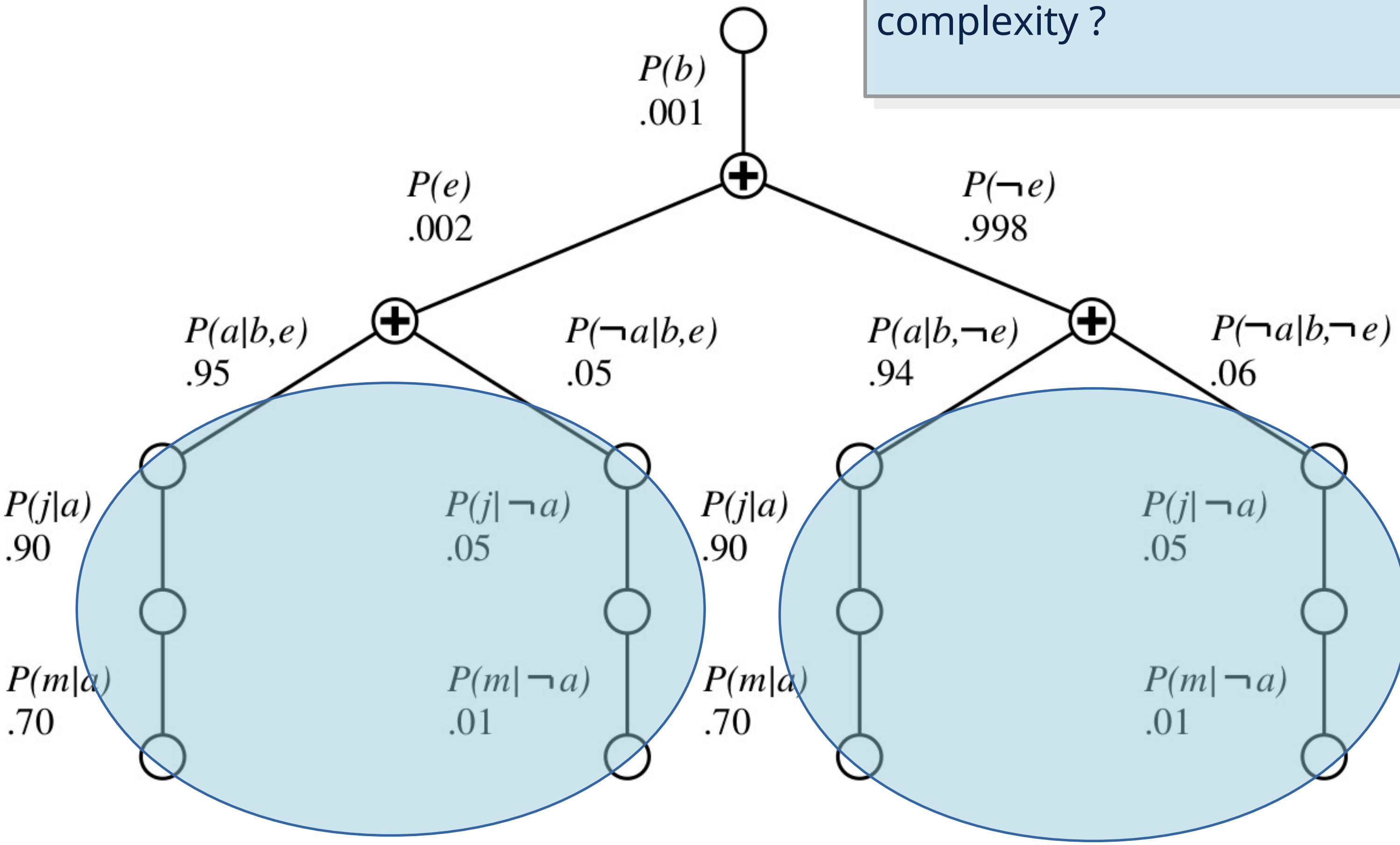


$$\mathbf{P}(B|j,m) = \alpha \mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e) P(j|a) P(m|a)$$

TU Delft

Enumeration Tree

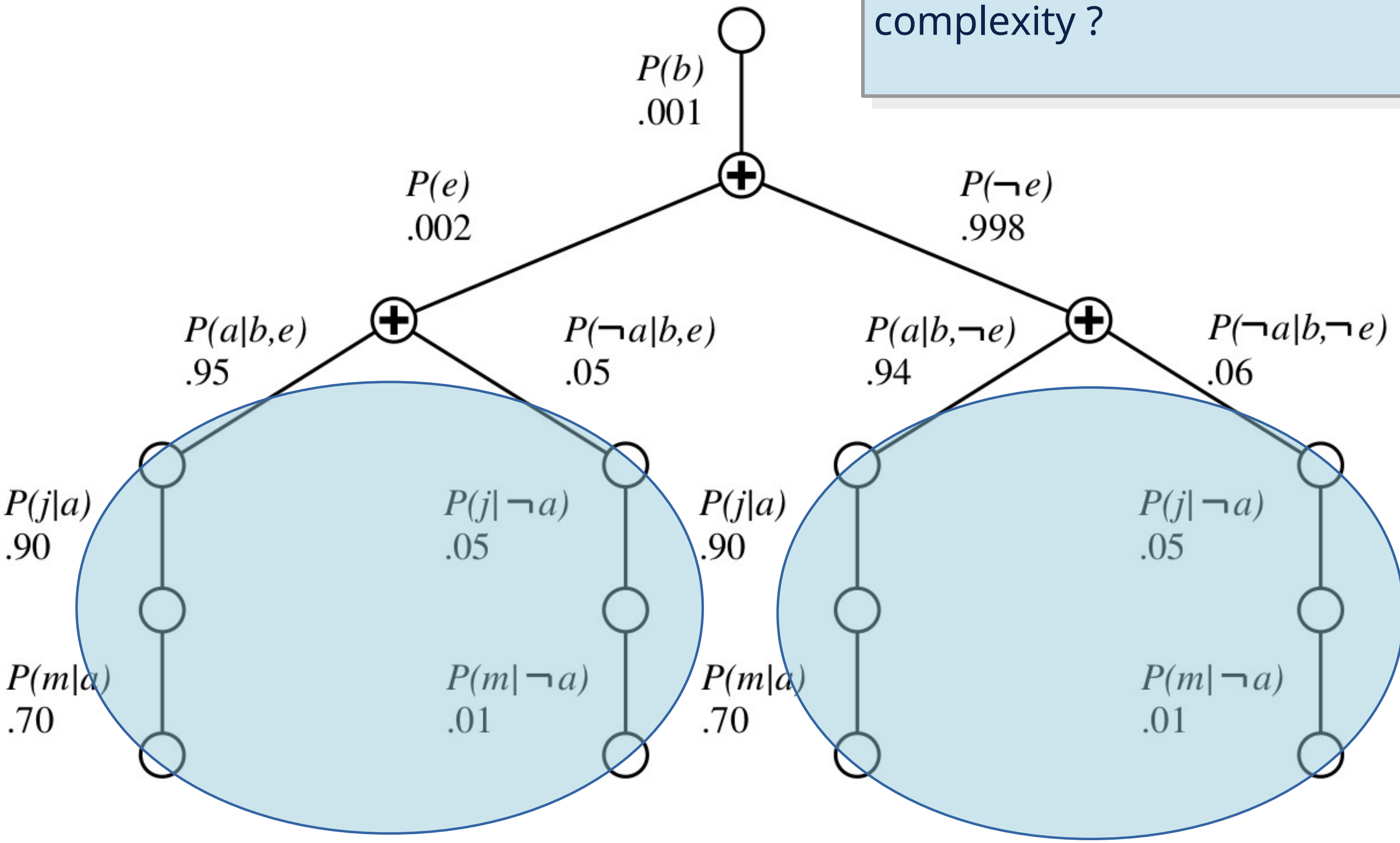
Can we improve upon $O(2^n)$ time complexity ?



$$\mathbf{P}(B|j,m) = \alpha \mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e) P(j|a) P(m|a)$$

Enumeration Tree

Can we improve upon $O(2^n)$ time complexity ?



$$\mathbf{P}(B|j,m) = \alpha \mathbf{P}(B) \sum_e P(e) \sum_a \mathbf{P}(a|B,e)P(j|a)P(m|a)$$

Perhaps we can **cache** these replicated computations...?

TUDelft

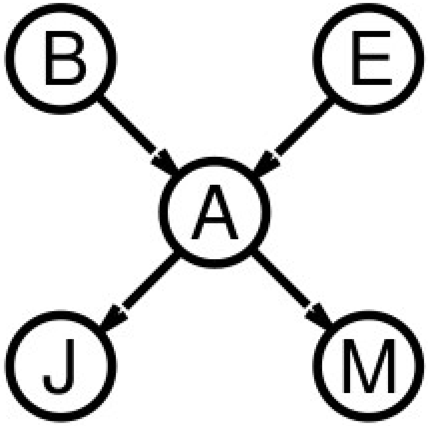
Variable Elimination (VE)

- First push in the summations as far as possible
- Then carry out summations **right-to-left**, caching intermediate results in new **factors**

$$
\begin{aligned}
\mathbf{P}(B|j,m) \\
&= \alpha \underbrace{\mathbf{P}(B)}_{B} \textstyle\sum_e \underbrace{P(e)}_{E} \textstyle\sum_a \underbrace{\mathbf{P}(a|B,e)}_{A} \underbrace{P(j|a)}_{J} \underbrace{P(m|a)}_{M} \\
&= \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a \mathbf{P}(a|B,e) P(j|a) f_M(a) \\
&= \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a \mathbf{P}(a|B,e) f_J(a) f_M(a) \\
&= \alpha \mathbf{P}(B) \textstyle\sum_e P(e) \textstyle\sum_a f_A(a,b,e) f_J(a) f_M(a) \\
&= \alpha \mathbf{P}(B) \textstyle\sum_e P(e) f_{\bar{A}JM}(b,e) \text{ (sum out } A\text{)} \\
&= \alpha \mathbf{P}(B) f_{\bar{E}\bar{A}JM}(b) \text{ (sum out } E\text{)} \\
&= \alpha f_B(b) \times f_{\bar{E}\bar{A}JM}(b)
\end{aligned}
$$

Variable Elimination (VE) – 2

- Implementation in terms of 2 operations:
  - pointwise-product: $\mathbf{f'}(A,B,C)=\mathbf{f}_1(A,B) \times \mathbf{f}_2(B,C)$
  - Sum-out: $\mathbf{f''}(A,B) = \sum_b \mathbf{f'}(A,b,C)$
- Complexity of VE: **time and space depends on largest factor constructed**.
  - **exponential** in the number of variables that participate in it.
  - trick: don't construct $\mathbf{f'}(A,B,C)$ explicitly,
compute entries $\mathbf{f'}(A,b,C)=\mathbf{f}_1(A,b) \times \mathbf{f}_2(b,C)$ "on-the-fly"
- How big is the largest factor?
  - depends on BN topology and picked 'ordering'
  - cannot bound in general....
- For **polytrees** VE is efficient:
  - runs in time and space linear in 'size' of BN.
  - polytree: between each pairs of nodes at most 1 undirected path



Are there better algorithms?

- Well, we can trade of time for space…
- But in terms of just time,there is little hope
  - Arnie Rosenthal (1977):

> then defined. It is shown that a variable-elimination procedure, nonserial dynamic programming, is optimal in an extremely strong sense among all algorithms in the subclass. The results' strong implications for choosing deterministic, adaptive, and nondeterministic algorithms for the optimization problem, for defining a complexity measure for a pattern of interactions, and for describing general classes of decomposition procedures are discussed. Several possible extensions and unsolved problems are mentioned.

- Exact inference is intractable (see R&N 14.4.3)

Wrap up

Uncertainty: Summary

- Agents need to represent beliefs
  - strong arguments: use probability
  - Bayes rule: to update beliefs

- Compact representations:
  Bayesian networks exploit conditional independence

- Exact inference:
  - enumeration / search
  - variable elimination
  - intractable in general, polytime on polytrees

- Next: Approximate inference, preferences & utilities

Uncertainty: Summary

- Agents need to represent beliefs
  - strong arguments: use probability
  - Bayes rule: to update beliefs
- Compact representations:
  Bayesian networks exploit condition
- Exact inference:
  - enumeration / search
  - variable elimination
  - intractable in general, polytime on polytrees
- Next: Approximate inference, preferences & utilities

next lecture might be fully online

check brighspace