Artificial Intelligence

12. Bayesian Networks

Florin Leon

"Gheorghe Asachi" Technical University of Iași
Faculty of Automatic Control and Computer Engineering

"Al. I. Cuza" University of Iași
Faculty of Computer Science

Bayesian Networks

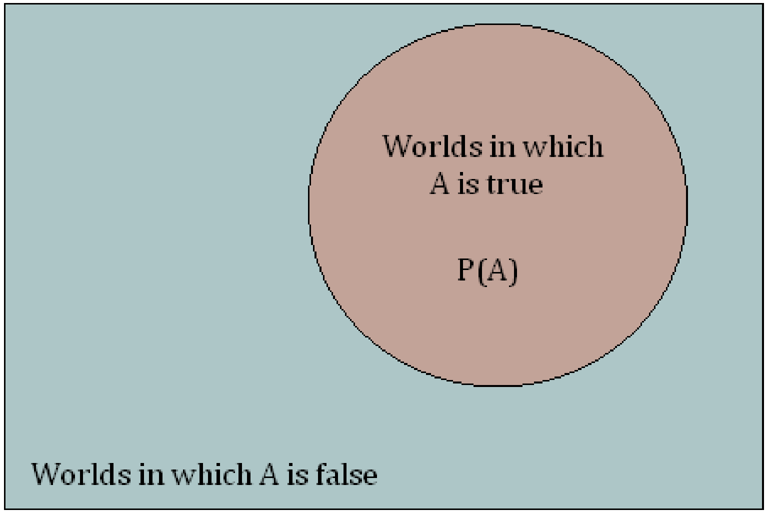
1. Probabilities
2. Bayesian Networks
3. Exact and Approximate Inferences
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

Bayesian Networks

1. Probabilities
   1.1. Bayes' Theorem
   1.2. Independence and conditional independence
2. Bayesian Networks
3. Exact and Approximate Inferences
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

Probabilities

- Frequentist interpretation (number of experiments)
  - $P(A)$ is the fraction of possible worlds where A is true



- Subjectivist interpretation (degrees of belief)

“Deterministic probabilities”

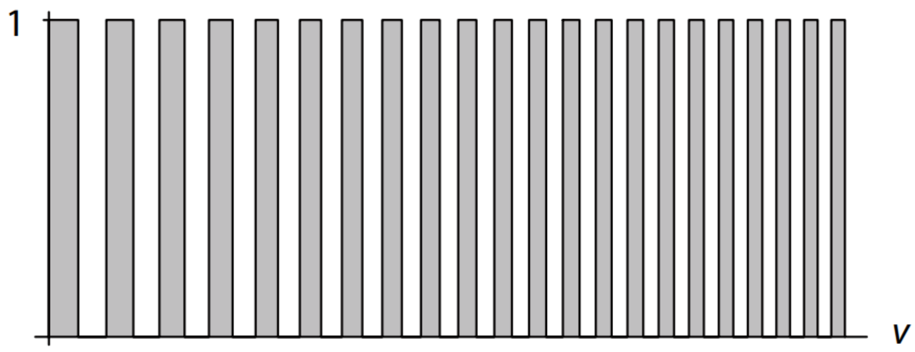


FIG. 1. Evolution function for a simple wheel of fortune, mapping initial spin speed v to either 1 (*red*) or 0 (*black*). The area under the function, corresponding to *red*-yielding values of v, is for clarity’s sake shaded gray.

“Deterministic probabilities”

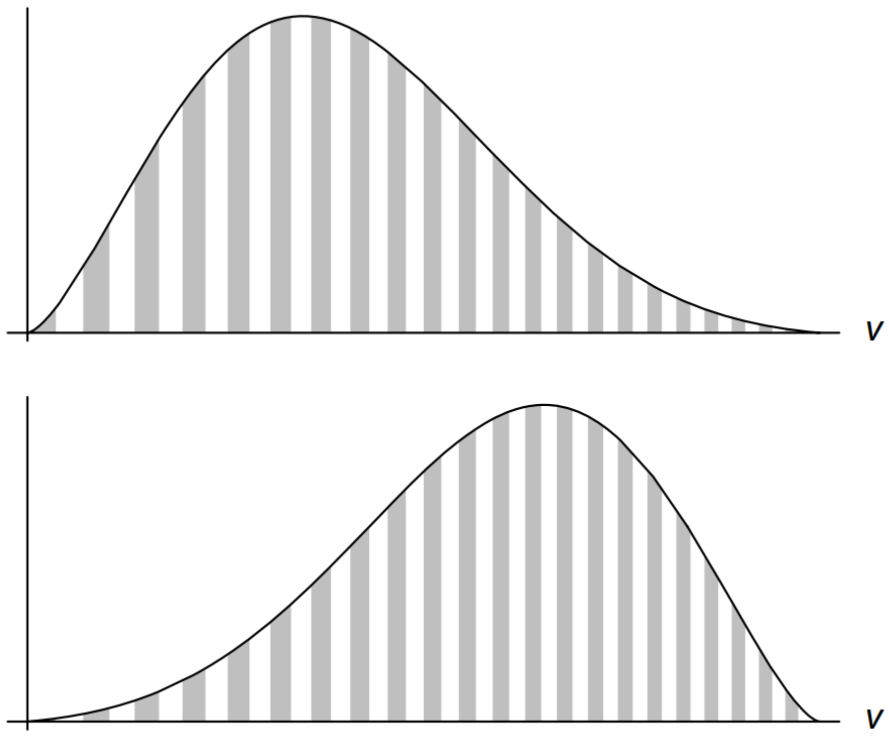


FIG. 2. Different spin speed distributions induce the same probability for *red*, equal to the strike ratio for *red* of one half.

“Deterministic probabilities”

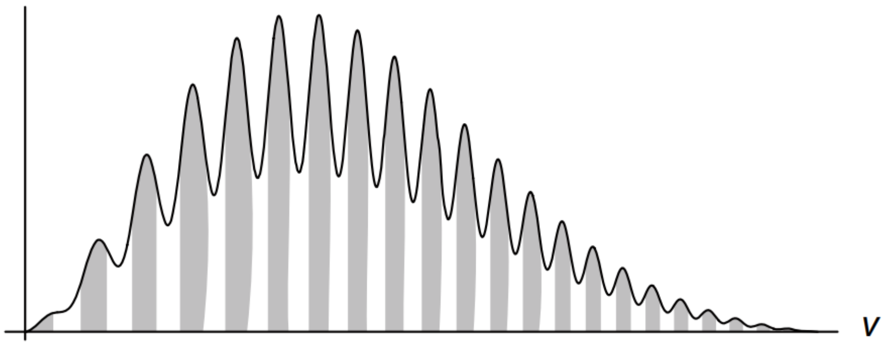
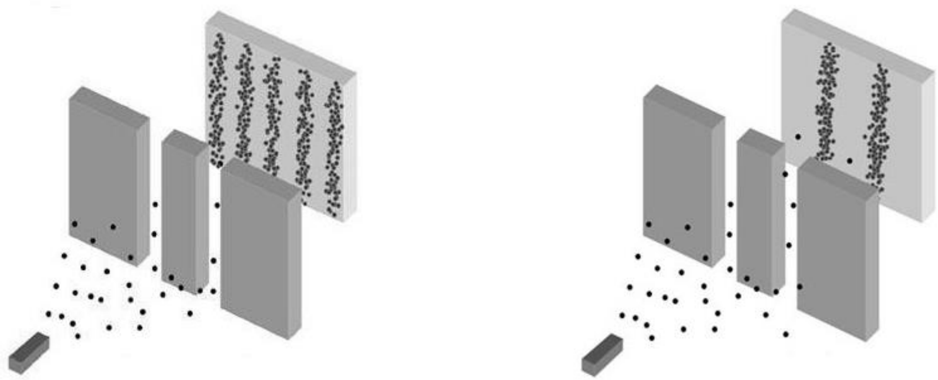


FIG. 3. Not every initial-spin-speed distribution induces a probability for *red* of one half.

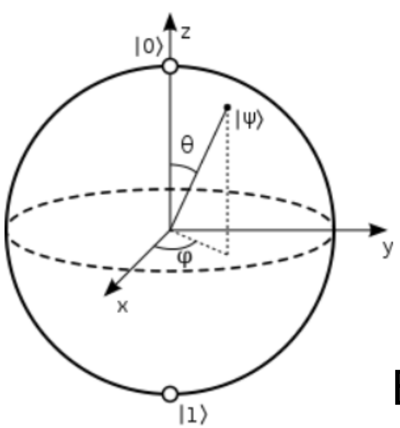
Probabilities

- Macroscopic world
  - Are probabilities only an expression of our inability to predict the evolution of complex processes?
- Quantum world
  - Unpredictable processes: radioactive decay
  - Double-slit experiment, quantum pigeonholes
  - Probability amplitudes
  - Quantum computing: qubits

The quantum world



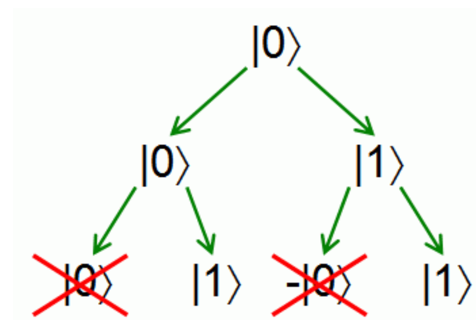
Double-slit experiment



$$|\Psi\rangle = a|0\rangle + b|1\rangle$$

Probability amplitudes

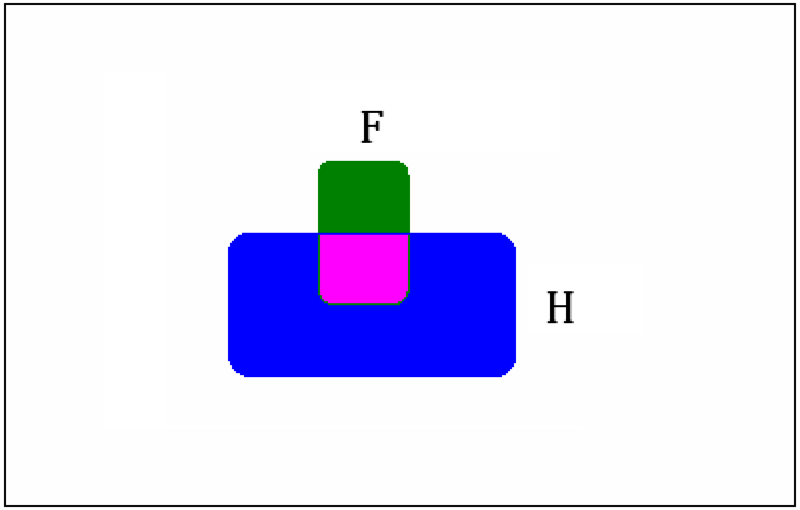
Bloch sphere



Interference with Hadamard gate

Conditional probabilities

- $P(A|B)$ is the fraction of possible worlds where B is true and then A is also true
  - The probability of A, given B



- H = headache, $P(H) = 1/10$
- F = flu, $P(F) = 1/40$
- $P(H|F) = 1/2$
- If someone has the flu, the probability of having a headache is 50%
- $P(H|F) = P(H \cap F) / P(F)$

Bayes' theorem

- $P(A|B) = P(A \cap B) \,/\, P(B)$
- $P(A \cap B) = P(A|B) \cdot P(B)$
- $P(A \cap B) = P(B|A) \cdot P(A)$

- $\Rightarrow \boxed{P(B|A) = P(A|B) \cdot P(B) \,/\, P(A)}$

posterior likelihood prior evidence

Bayes’ theorem

- $P(B|A) = P(A|B) \cdot P(B) / P(A)$

- Thomas Bayes (1763). *An essay towards solving a problem in the doctrine of chances*. Philosophical Transactions of the Royal Society of London, vol. 53, pp. 370-418

Bayes' theorem

- $P(H|E) = P(E|H) \cdot P(H) \,/\, P(E)$
  - H = hypothesis, cause
  - E = evidence, effect

Diagnosis

- Known probabilities
  - **M**eningitis: $P(M) = 0.002\%$
  - **S**tiff neck: $P(S) = 5\%$
  - Meningitis causes a stiff neck in half of the cases: $P(S|M) = 50\%$
- If a patient has a stiff neck, which is the probability of meningitis?
  - $P(M|S) = P(S|M) \cdot P(M) / P(S) = 0.02\%$

Diagnosis

- A mistake people sometimes make:
  $P(A|B) = P(B|A)$
- Diagnostics for rare diseases
  - One must take into account the probability of the tests to return false positives

$$P(D) = 0.01$$

$$P(T|D) = 0.99$$

$$P(T|\neg D) = 0.02$$

D – disease
T – test

$$P(D|T) = \frac{P(T|D) \cdot P(D)}{P(T)} = \frac{P(T|D) \cdot P(D)}{P(T|D) \cdot P(D) + P(T|\neg D) \cdot P(\neg D)} = 0.33$$

Independence and conditional independence

- Example 1. John and Mary flip a coin 100 times. Each of them has a different coin
  - Independent events
  - The result of an experiment does not influence the result of the other experiment
  - Knowing about an experiment does not provide new information about the other

Independence and conditional independence

- Example 2a. John and Mary flip the same coin
  - If the coin is biased, event *A* (John) can provide information about event *B* (Mary)
  - The events are not independent
  - The result of an experiment can influence knowledge about the result of the other

Independence and conditional independence

- Example 2b. Let C be the variable “the coin is biased towards tails”, an expert’s opinion
  - If we know C, experiment A no longer provides new information about B
    - $P(B|A,C) = P(B|C)$
  - A and B are conditionally independent given C
  - The situation is known as the “common cause”

Independence and conditional independence

- Example 3. John and Mary live in different areas of the city and come to work by tram and by car, respectively
  - “John is late” and “Mary is late” can be considered to be independent
  - If the tram drivers are on strike, then the car traffic increases. The events are *conditionally* independent
- There are many real life situations when events considered to be independent are in fact only conditionally independent

Independence and conditional independence

- Example 4. Both a cold and an allergy can make John sneeze
  - If we do not know that John sneezed, cold and allergy are independent
  - If we know that John sneezed, cold and allergy are no longer independent
  - An increase in the probability of a cold decreases the probability of allergy and vice versa
  - The situation is known as "explaining away"

Joint probability distribution

- 3 binary variables: $2^3 - 1 = 7$ independent parameters

x	y	z	$p(x,y,z)$
0	0	0	0.12
0	0	1	0.18
0	1	0	0.04
0	1	1	0.16
1	0	0	0.09
1	0	1	0.21
1	1	0	0.02
1	1	1	0.18

← 1 – sum of others

Representing uncertain knowledge

- A situation with 5 variables (next example)
  - Specifies a full joint probability distribution with $2^5 - 1 = 31$ parameters
    - Feasible
- An expert system with 37 variables for monitoring intensive care patients
  - $2^{37} - 1 \approx 10^{11}$ parameters
    - Unfeasible

Representing the full joint distribution

$$P(x_1, \ldots, x_n) = P(x_n | x_{n-1}, \ldots, x_1) P(x_{n-1}, \ldots, x_1)$$

$$\begin{aligned} P(x_1, \ldots, x_n) &= P(x_n | x_{n-1}, \ldots, x_1) P(x_{n-1} | x_{n-2}, \ldots, x_1) \cdots P(x_2 | x_1) P(x_1) \\ &= \prod_{i=1}^{n} P(x_i | x_{i-1}, \ldots, x_1) . \end{aligned}$$

Chain rule

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

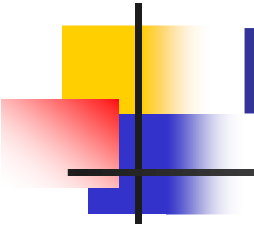
True only if every node is conditionally independent of its ancestors in the ordered series of nodes, given its parents

$$\mathbf{P}(Cause, Effect_1, \ldots, Effect_n) = \mathbf{P}(Cause) \prod_i \mathbf{P}(Effect_i | Cause)$$

If the effects are considered independent ⇒ *Naïve Bayes*

Bayesian Networks

1. Probabilities
2. Bayesian Networks
    2.1. The Bayes-Ball algorithm
3. Exact and Approximate Inferences
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

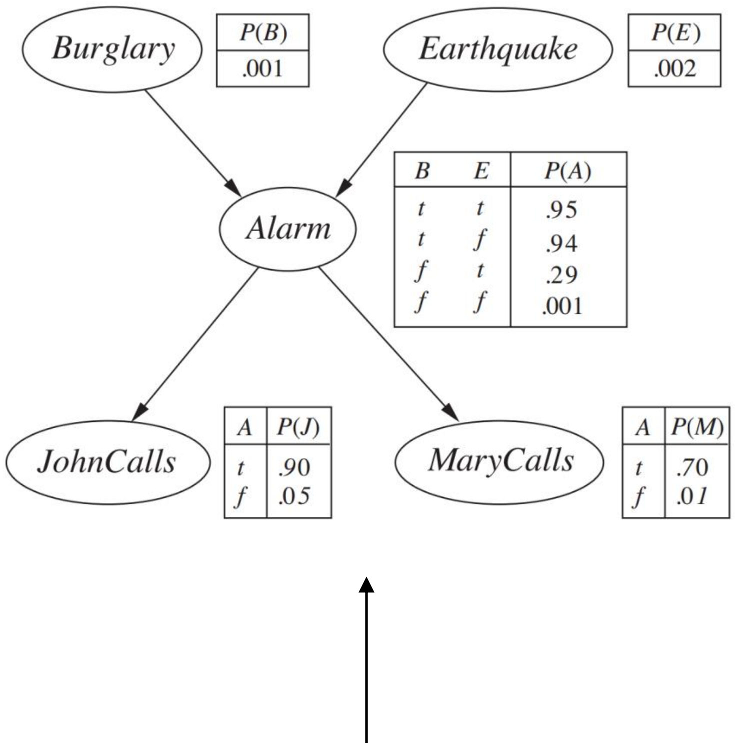



Bayesian network

- A Bayesian network is a probabilistic graphical model, i.e. a graph with a set of nodes, which represent random events, connected by arcs, which represent conditional dependencies between events

Example

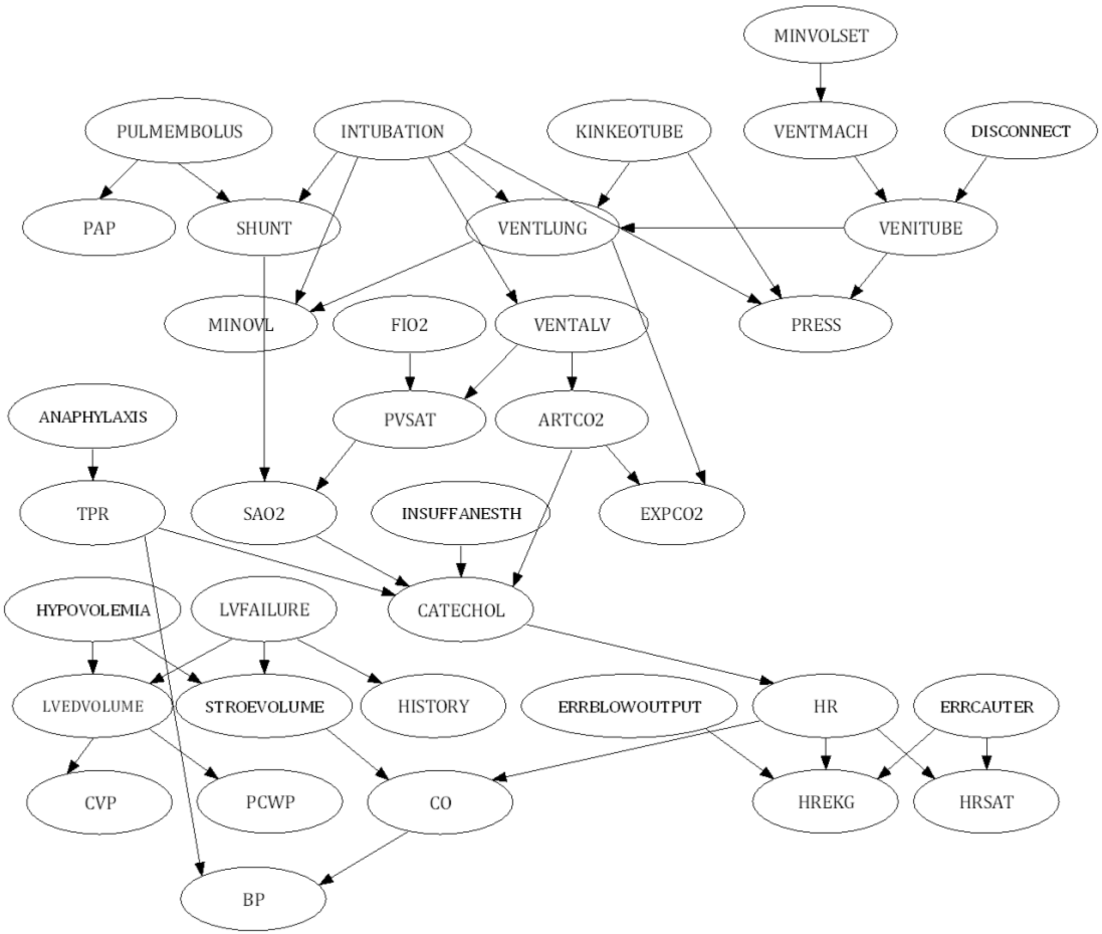
- A new alarm system has been installed. It sounds when a burglary occurs, but also in case of an earthquake
- The neighbours John and Mary call the owner at work if they hear the alarm
- 10 independent parameters instead of 31



Bayesian network

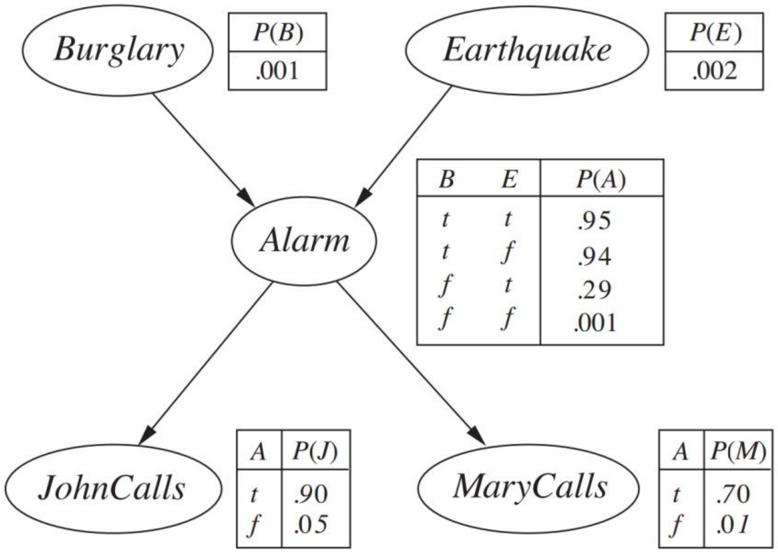
Comparison

- Expert system for monitoring intensive care patients
- 37 variables
- 509 parameters instead of 10^{11} (100,000,000,000)



Simple queries

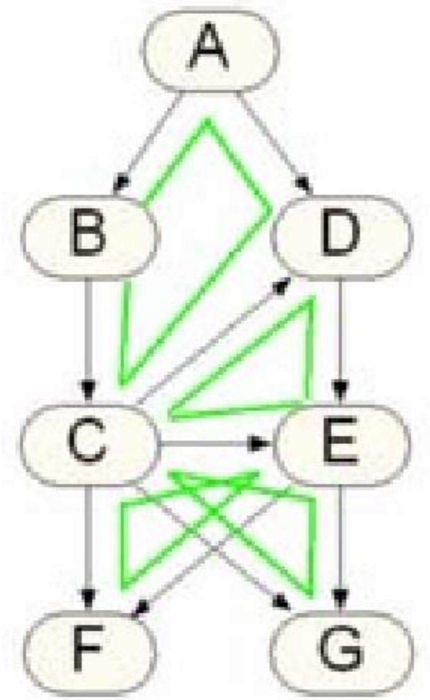
- *What is the probability that the alarm has sounded but neither a burglary nor an earthquake has occurred, and still both John and Mary call?*

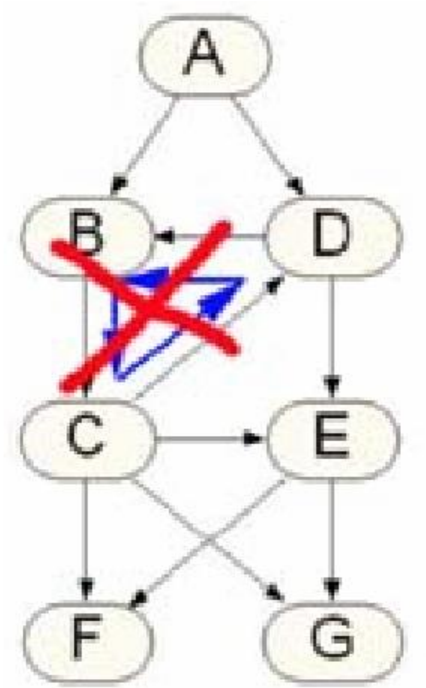


$$
\begin{aligned}
P(j, m, a, \neg b, \neg e) &= P(j \mid a) P(m \mid a) P(a \mid \neg b \wedge \neg e) P(\neg b) P(\neg e) \\
&= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 = 0.000628
\end{aligned}
$$

The validity of a Bayesian network

- A Bayesian network is a directed acyclic graph
- The arcs cannot form directed loops

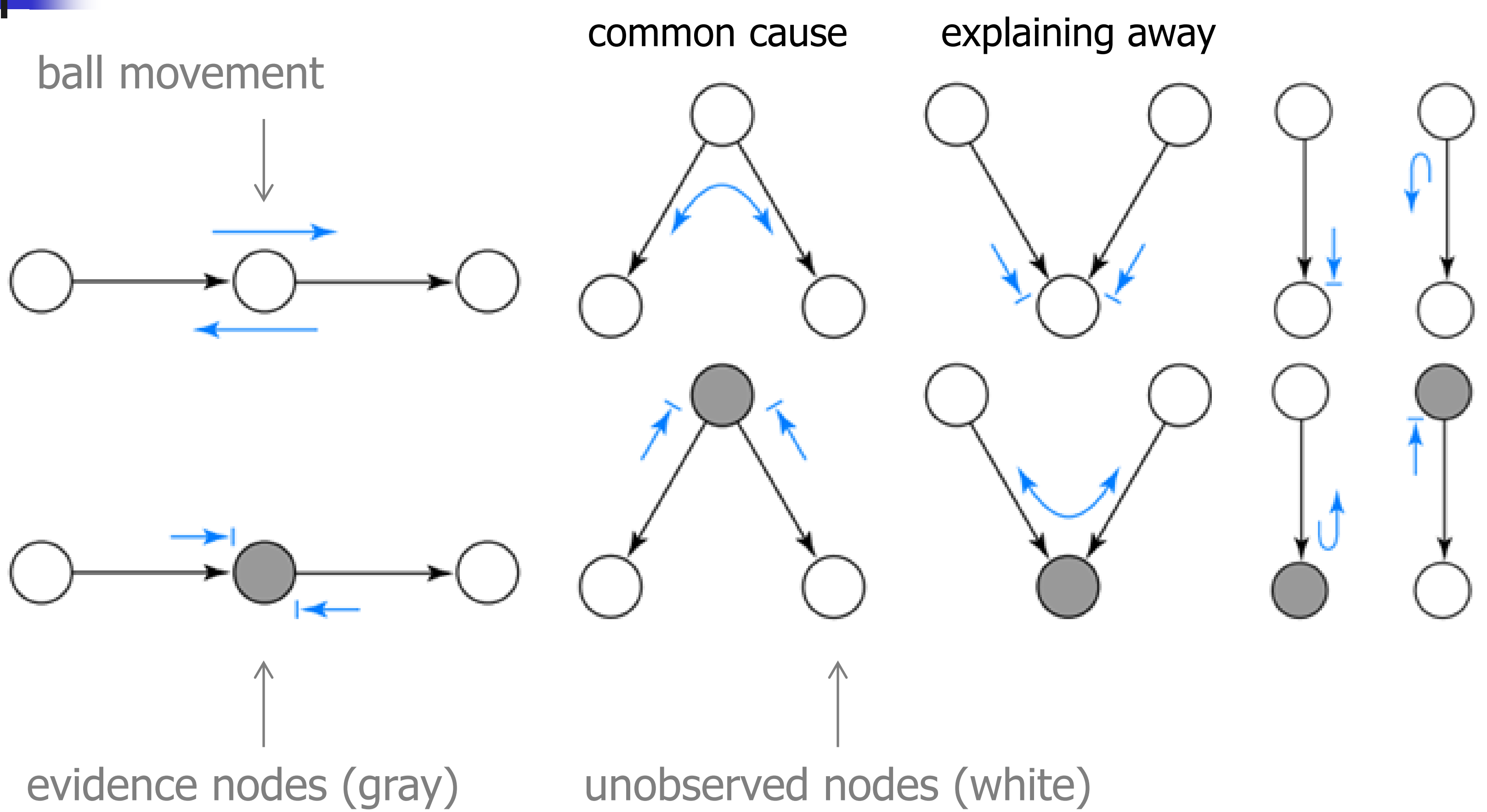




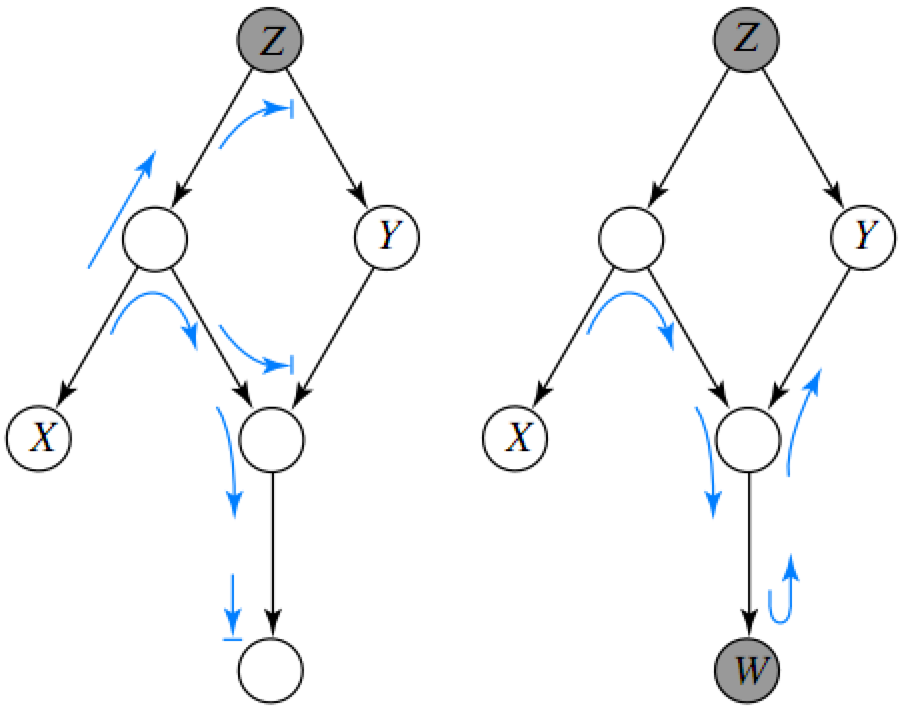
The Bayes-Ball algorithm

- A simple way to determine the independence and conditional independence relations in a Bayesian network
- It assumes sending a ball from a node into the network
- It passes in different ways, according to who passes it (child or parent) and the state of the node that receives the ball (observed/evidence or unobserved)
- The nodes not reached by the ball are (conditionally) independent of the start node

Rules for passing the ball



Examples



no active paths

$X \perp\!\!\!\perp Y \mid Z$

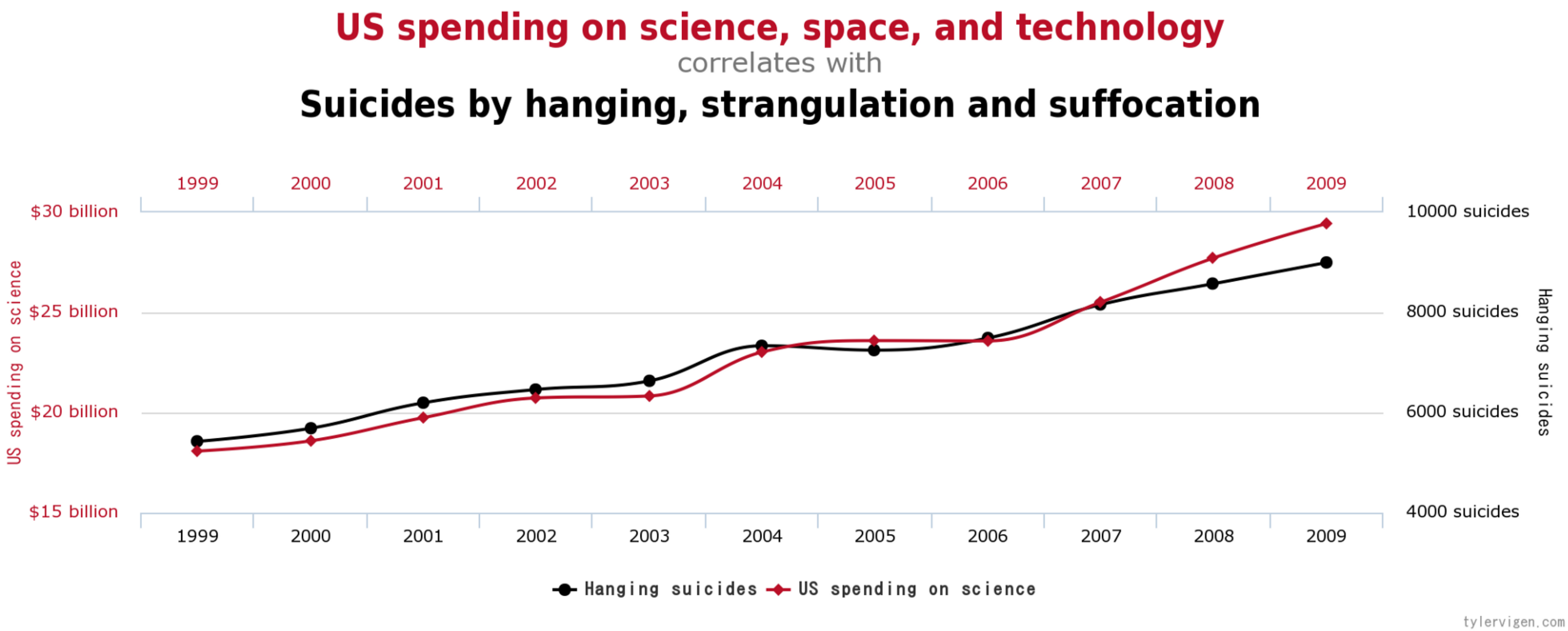
one active path

$X \not\perp\!\!\!\perp Y \mid \{W, Z\}$

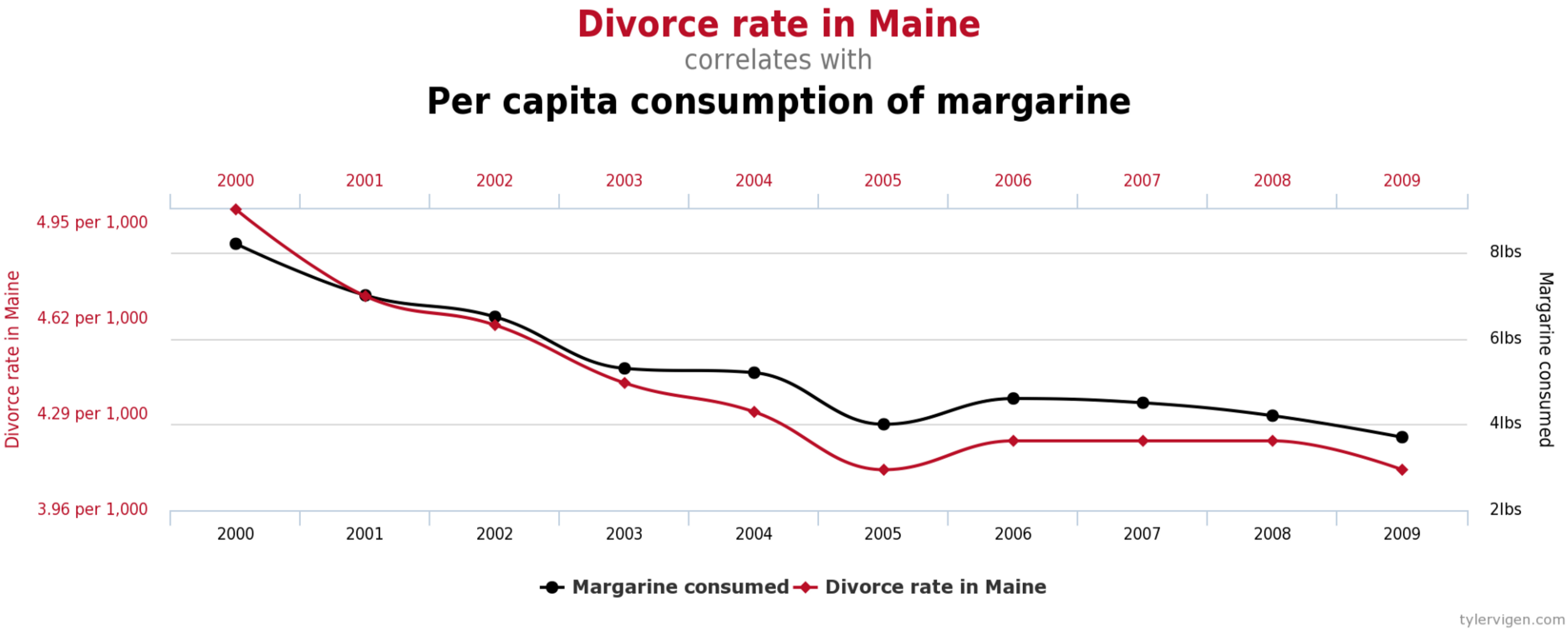
Correlation and causality

- Bayesian networks encode correlation relationships
  - The probability of an event A varies with the probabilities of other events $\{B, C, ...\}$
  - A is correlated with $\{B, C, ...\}$
  - A may or may not be caused by $\{B, C, ...\}$
  - All events $\{A, B, C, ...\}$ could be caused by other unknown events
- Correlation $\not\Rightarrow$ causality
- Causality $\Rightarrow$ correlation

Strange correlations



Strange correlations



Bayesian Networks

1. Probabilities
2. Bayesian Networks
3. Exact and Approximate Inferences
    3.1. Inference of marginal probabilities
    3.2. Inference by enumeration
    3.3. Inference by likelihood weighting
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

Topological sorting

- The topological sorting of a graph is a linear ordering such as, for each arc $A \rightarrow B$, A apears before B
- For a Bayesian network, the topological sorting ensures that the parent nodes will appear before the child nodes
- Algorithms usually have a linear time complexity in the number of nodes n and arcs a: $O(n + a)$
- There are several topological sorting algorithms
- If the graph is directed acyclic, there is at least one solution
- If there are cycles in the graph, topological sorting is impossible

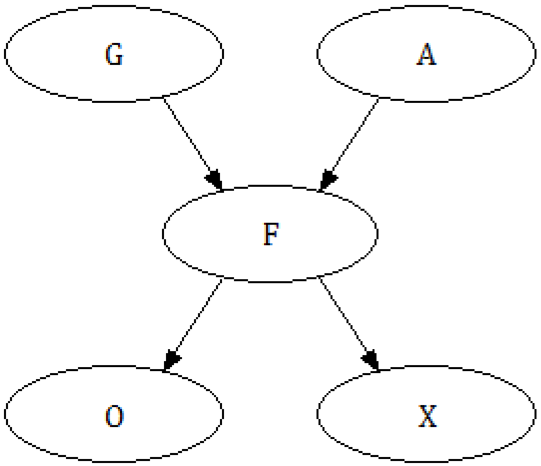
Kahn's algorithm

```
L ← an initially empty list that will contain sorted items
S ← the set of nodes without parents

while S is not empty
   remove a node n from S
   add n to L
   foreach node m with an arc e from n to m
      remove arc e from the graph
      if m has no other incident arcs then
         add m to S

if the graph has more arcs then
   return error(the graph has at least one cycle)
else
   return L (the topologically sorted items)
```

Example



1. $L = \emptyset$, $S = \{G, A\}$

2. $L = \{G\}$, $S = \{A\}$

3. se elimină arcul *GF*, *F* nu poate fi adăugat în *S* pentru că mai există arcul *AF*

4. $L = \{G, A\}$, $S = \emptyset$

5. se elimină arcul *AF*, $S = \{F\}$

6. $L = \{G, A, F\}$, $S = \emptyset$

7. se elimină arcul *FO*, $S = \{O\}$

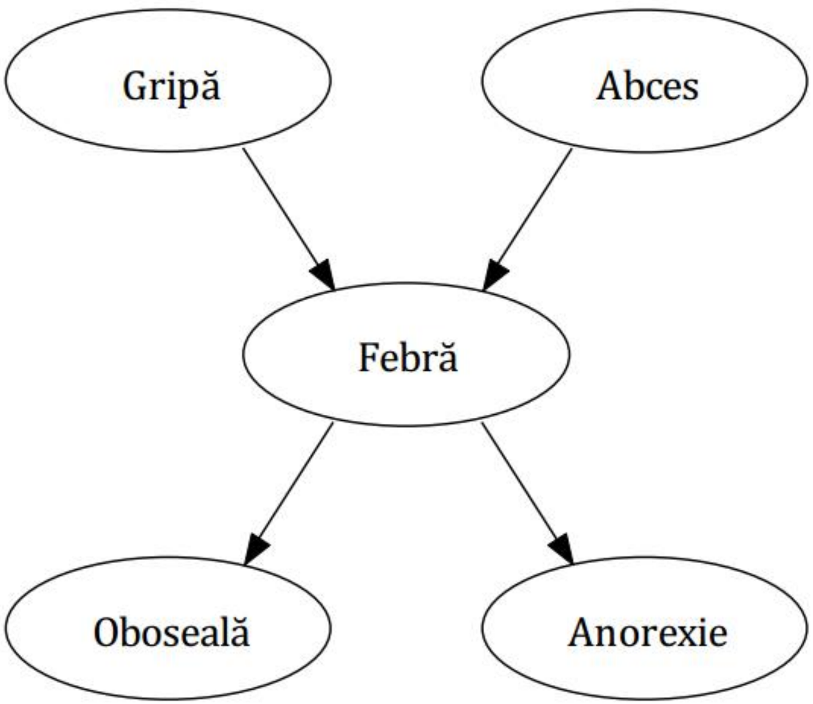
8. se elimină arcul *FX*, $S = \{O, X\}$

9. $L = \{G, A, F, O\}$, $S = \{X\}$

10. $L = \{G, A, F, O, X\}$, $S = \emptyset$

Soluția este deci: $\{G, A, F, O, X\}$.

Computing the marginal probabilities



P(*Gripă* = *Da*)	*P*(*Gripă* = *Nu*)
0,1	0,9

P(*Abces* = *Da*)	*P*(*Abces* = *Nu*)
0,05	0,95

Gripă	*Abces*	*P*(*Febră* = *Da*)	*P*(*Febră* = *Nu*)
Da	Da	0,8	0,2
Da	Nu	0,7	0,3
Nu	Da	0,25	0,75
Nu	Nu	0,05	0,95

Febră	*P*(*Oboseală* = *Da*)	*P*(*Oboseală* = *Nu*)
Da	0,6	0,4
Nu	0,2	0,8

Febră	*P*(*Anorexie* = *Da*)	*P*(*Anorexie* = *Nu*)
Da	0,5	0,5
Nu	0,1	0,9

Node *F*

$$P(F_D) =$$

$$P(F_D|G_D, A_D) \cdot P(G_D) \cdot P(A_D) +$$

$$P(F_D|G_D, A_N) \cdot P(G_D) \cdot P(A_N) +$$

$$P(F_D|G_N, A_D) \cdot P(G_N) \cdot P(A_D) +$$

$$P(F_D|G_N, A_N) \cdot P(G_N) \cdot P(A_N) =$$

$$0{,}8 \cdot 0{,}1 \cdot 0{,}05 + 0{,}7 \cdot 0{,}1 \cdot 0{,}95 + 0{,}25 \cdot 0{,}9 \cdot 0{,}05 + 0{,}05 \cdot 0{,}9 \cdot 0{,}95 = 0{,}1245 \approx 12\%.$$

$$P(F_N) = 1 - P(F_D) = 0{,}8755 \approx 88\%.$$

F_D = (Febră = Da)
F_N = (Febră = Nu)

In a Bayesian network, a node can have as many values as possible, not just two

Node *O*

$$P(O_D) =$$

$$P(O_D|F_D) \cdot P(F_D) + P(O_D|F_N) \cdot P(F_N) =$$

$$0{,}6 \cdot 0{,}1245 + 0{,}2 \cdot 0{,}8755 = 0{,}2498 \approx 25\%,$$

$$P(O_N) = 1 - P(O_D) = 0{,}7502 \approx 75\%.$$

O_D = (*Oboseală* = *Da*)
O_N = (*Oboseală* = *Nu*)

Node *X*

$$P(X_D) =$$

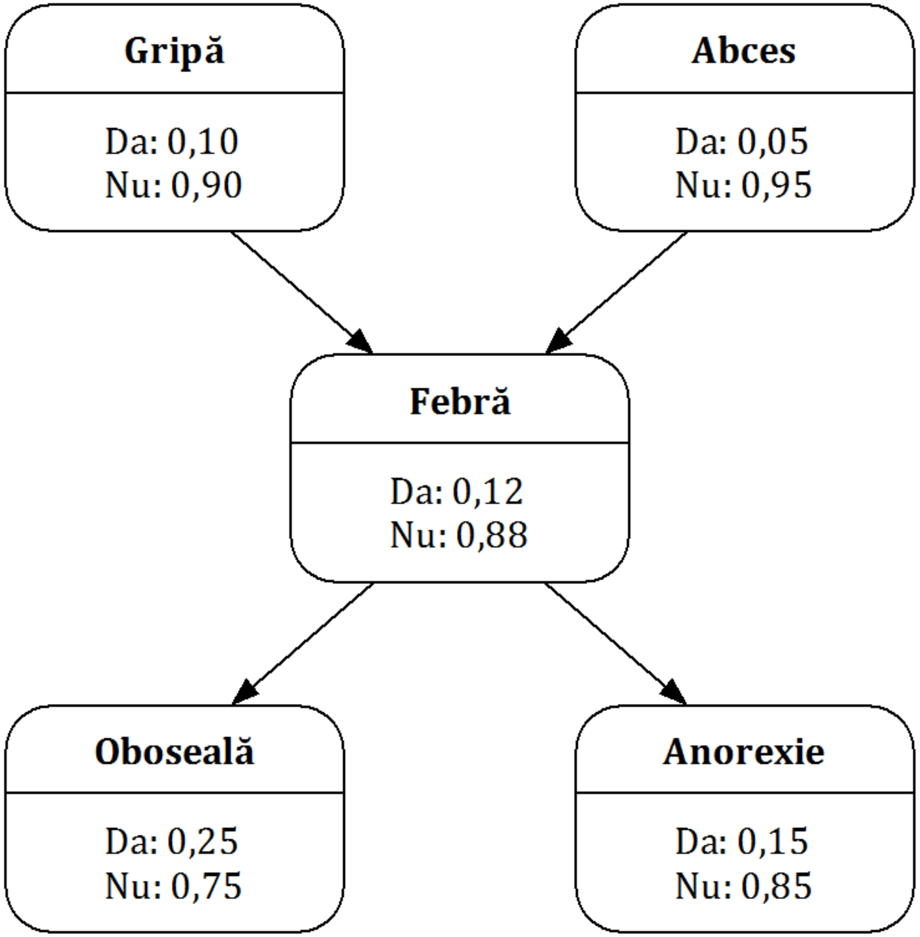
$$P(X_D|F_D) \cdot P(F_D) + P(X_D|F_N) \cdot P(F_N) =$$

$$0{,}5 \cdot 0{,}1245 + 0{,}1 \cdot 0{,}8755 = 0{,}1498 \approx 15\%,$$

$$P(X_N) = 1 - P(X_D) = 0{,}8502 \approx 85\%.$$

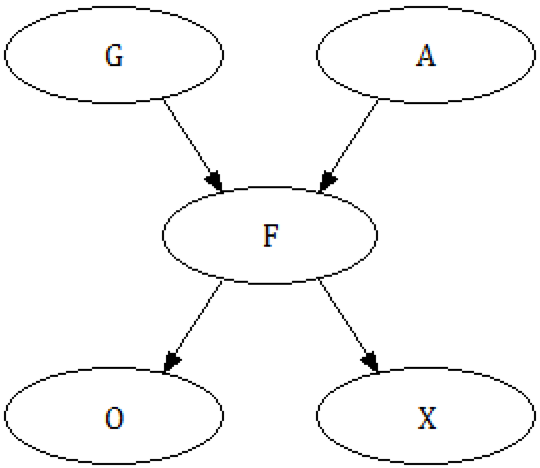
X_D = (*Anorexie* = *Da*)
X_N = (*Anorexie* = *Nu*)

Marginal probabilities of nodes



Inference by enumeration

- Query: *What is the probability that a person has the flu (G) if he/she shows symptoms of fatigue (O) and anorexia (X)?*



Solution

Normalization coefficient

$$\mathbf{P}(X|\mathbf{e}) = \alpha\,\mathbf{P}(X,\mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X,\mathbf{e},\mathbf{y})$$

Queried variable

Observed evidence

Unobserved variables

Sum over all possible values of **y**,
e.g. *true* or *false*

Solution

- We compute $P(G_D|O_D, X_D)$ and $P(G_N|O_D, X_D)$ independently
- For $P(G_D|O_D, X_D)$, the remaining variables are *A* and *F*
- We sum the probabilities of all values for these variables : $a \in \{A_D, A_N\}$ and $f \in \{F_D, F_N\}$
- To increase the efficiency of the calculations, it is recommended that the remaining variables be sorted topologically first so that parents appear before their children
- In this case, the sums can be decomposed more easily, moving the factors that do not depend on certain variables outside those summations

Solution

$$P(G_D|O_D,X_D) =$$

$$\alpha \cdot \sum_{a\in\{A_D,A_N\}} \sum_{f\in\{F_D,F_N\}} P(G_D,a,f,O_D,X_D) =$$

$$\alpha \cdot \sum_a \sum_f P(G_D) \cdot P(a) \cdot P(f|G_D,a) \cdot P(O_D|f) \cdot P(X_D|f) =$$

$$\alpha \cdot P(G_D) \cdot \sum_a P(a) \cdot \sum_f P(f|G_D,a) \cdot P(O_D|f) \cdot P(X_D|f) =$$

$$\alpha \cdot P(G_D) \cdot \sum_a P(a) \cdot [P(F_D|G_D,a) \cdot P(O_D|F_D) \cdot P(X_D|F_D) +$$

$$P(F_N|G_D,a) \cdot P(O_D|F_N) \cdot P(X_D|F_N)] =$$

$$\alpha \cdot P(G_D) \cdot \{P(A_D) \cdot [P(F_D|G_D,A_D) \cdot P(O_D|F_D) \cdot P(X_D|F_D) +$$

$$P(F_N|G_D,A_D) \cdot P(O_D|F_N) \cdot P(X_D|F_N)] +$$

$$P(A_N) \cdot [P(F_D|G_D,A_N) \cdot P(O_D|F_D) \cdot P(X_D|F_D) +$$

$$P(F_N|G_D,A_N) \cdot P(O_D|F_N) \cdot P(X_D|F_N)]\} =$$

$$\alpha \cdot 0{,}1 \cdot \{0{,}05 \cdot [0{,}8 \cdot 0{,}6 \cdot 0{,}5 + 0{,}2 \cdot 0{,}2 \cdot 0{,}1] +$$

$$0{,}95 \cdot [0{,}7 \cdot 0{,}6 \cdot 0{,}5 + 0{,}3 \cdot 0{,}2 \cdot 0{,}1]\} =$$

$$\alpha \cdot 0{,}02174.$$

P(Gripă = Da)	*P(Gripă = Nu)*
0,1	0,9

P(Abces = Da)	*P(Abces = Nu)*
0,05	0,95

Gripă	*Abces*	*P(Febră = Da)*	*P(Febră = Nu)*
Da	Da	0,8	0,2
Da	Nu	0,7	0,3
Nu	Da	0,25	0,75
Nu	Nu	0,05	0,95

Febră	*P(Oboseală = Da)*	*P(Oboseală = Nu)*
Da	0,6	0,4
Nu	0,2	0,8

Febră	*P(Anorexie = Da)*	*P(Anorexie = Nu)*
Da	0,5	0,5
Nu	0,1	0,9

Solution

$$P(G_N|O_D,X_D) =$$

$$\alpha \cdot \sum_{a\in\{A_D,A_N\}} \sum_{f\in\{F_D,F_N\}} P(G_N,a,f,O_D,X_D) =$$

$$\alpha \cdot \sum_a \sum_f P(G_N) \cdot P(a) \cdot P(f|G_N,a) \cdot P(O_D|f) \cdot P(X_D|f) =$$

$$\alpha \cdot P(G_N) \cdot \sum_a P(a) \cdot \sum_f P(f|G_N,a) \cdot P(O_D|f) \cdot P(X_D|f) =$$

$$\alpha \cdot P(G_N) \cdot \{P(A_D) \cdot [P(F_D|G_N,A_D) \cdot P(O_D|F_D) \cdot P(X_D|F_D) +$$
$$P(F_N|G_N,A_D) \cdot P(O_D|F_N) \cdot P(X_D|F_N)] +$$
$$P(A_N) \cdot [P(F_D|G_N,A_N) \cdot P(O_D|F_D) \cdot P(X_D|F_D) +$$
$$P(F_N|G_N,A_N) \cdot P(O_D|F_N) \cdot P(X_D|F_N)]\} =$$

$$\alpha \cdot 0{,}9 \cdot \{0{,}05 \cdot [0{,}25 \cdot 0{,}6 \cdot 0{,}5 + 0{,}75 \cdot 0{,}2 \cdot 0{,}1] +$$
$$0{,}95 \cdot [0{,}05 \cdot 0{,}6 \cdot 0{,}5 + 0{,}95 \cdot 0{,}2 \cdot 0{,}1]\} =$$

$$\alpha \cdot 0{,}03312.$$

P(Gripă = Da)	*P(Gripă = Nu)*
0,1	0,9

P(Abces = Da)	*P(Abces = Nu)*
0,05	0,95

Gripă	*Abces*	*P(Febră = Da)*	*P(Febră = Nu)*
Da	Da	0,8	0,2
Da	Nu	0,7	0,3
Nu	Da	0,25	0,75
Nu	Nu	0,05	0,95

Febră	*P(Oboseală = Da)*	*P(Oboseală = Nu)*
Da	0,6	0,4
Nu	0,2	0,8

Febră	*P(Anorexie = Da)*	*P(Anorexie = Nu)*
Da	0,5	0,5
Nu	0,1	0,9

Result

- $P(G_D|O_D, X_D) + P(G_N|O_D, X_D) = 1$
- $P(G_D|O_D, X_D) = \alpha \cdot 0.02174$
- $P(G_N|O_D, X_D) = \alpha \cdot 0.03312$
- $\Rightarrow \alpha = 18.23$

- $P(G_D|O_D, X_D) = 0.39628 \approx 40\%$
- $P(G_N|O_D, X_D) = 0.60372 \approx 60\%$

Pseudocode

function ENUMERATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for some set of variables **E**
    bn, a Bayes net
  Q ← a distribution over X, where $\mathbf{Q}(x_i)$ is $P(X=x_i)$
  for each value x_i that X can have **do**
    $\mathbf{Q}(x_i)$ ← ENUMERATE-ALL(bn.VARS, $\mathbf{e}_{x_i}$), where $\mathbf{e}_{x_i}$ is the evidence **e** plus the assignment $X=x_i$
  return NORMALIZE(**Q**)

function ENUMERATE-ALL($vars$, **e**) **returns** a probability (a real number in [0,1])
  inputs: $vars$, a list of all the variables
    e, observed values for some set of variables **E**
  if EMPTY($vars$) **then return** 1.0
  Y ← FIRST($vars$)
  if Y is assigned a value (call it y) in **e then**
    return $P(Y=y \mid$ values assigned to Y's parents in $\mathbf{e}) \times$ ENUMERATE-ALL(REST($vars$), **e**)
  else
    return $\sum_{y_i}$ [$P(Y=y_i \mid$ values assigned to Y's parents in $\mathbf{e}) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_{y_i}$)],
      where $\mathbf{e}_{y_i}$ is the evidence **e** plus the assignment $Y=y_i$

Approximate inference

- For “real world” problems, Bayesian networks with hundreds of nodes have been built, for which exact algorithms reach their limits, because inference is an NP-hard problem
- For very complex networks, the approximate inference is the only possibility to obtain a result
- For other problems where accuracy is not a critical factor, approximate inference can greatly increase the calculation speed

Stochastic inference by likelihood weighting

- Samples / instantiations of the network are generated randomly and the desired probabilities are calculated as relative occurrence frequencies
- The nodes without parents will be instantiated according to their marginal probabilities
- The evidence nodes always take the observed values
- The values of the unobserved variables have probabilities of occurrence according to the probabilities of the nodes

Procedure

- For each sampling *s* of the network, a weight is calculated:

$$w(s) = \frac{\prod_{x \in U} P\left(x \middle| \pi(x)\right)}{\prod_{x \in U \setminus E} P\left(x \middle| \pi(x)\right)}$$

- where *U* is the set of all nodes, *E* is the set of evidence nodes, and $\pi(x)$ are the parents of node *x*
- The process is repeated for a preset number of samples

Procedure

- Finally, a normalization phase takes place, in which the sum of the weights of cases is calculated where a query variable had a certain value, divided by the sum of the weights of all cases:

$$P(Var = val) = \frac{w_T}{w_S} = \frac{\sum_{s \in T} w(s)}{\sum_{s \in S} w(s)}$$

- where S is the set of all samples, and $T \subseteq S$ is the subset of samples where variable *Var* appears with value *val*

Example: the same problem

- Query variable: *G*
- Evidence variables: O_D, X_D
- Unobserved variables: *A*, *F*
- Topological sorting: *G, A, F, O, X*
- The variables receive values at random, according to the probabilities in the tables
- Notation: $A \sim M(Yes : 0.1, No : 0.9)$ means that variable *A* will have value *Yes* with probability 10% and *No* with probability 90%
  - $M(\cdot)$ = the *Multinoulli* / generalized Bernoulli / categorical distribution ~ *roulette wheel selection*

Sampling 1

- $G \sim M(Da : 0.1, Nu : 0.9) \Rightarrow G_N$
- $A \sim M(Da : 0.05, Nu : 0.95) \Rightarrow A_D$
- $F \sim M(Da : 0.25, Nu : 0.75) \Rightarrow F_N$
- O_D
- X_D
- $s_1 = (G_N, A_D, F_N, O_D, X_D)$

P(Gripă = Da)	P(Gripă = Nu)
0,1	0,9

P(Abces = Da)	P(Abces = Nu)
0,05	0,95

Gripă	Abces	P(Febră = Da)	P(Febră = Nu)
Da	Da	0,8	0,2
Da	Nu	0,7	0,3
Nu	Da	0,25	0,75
Nu	Nu	0,05	0,95

Febră	P(Oboseală = Da)	P(Oboseală = Nu)
Da	0,6	0,4
Nu	0,2	0,8

Febră	P(Anorexie = Da)	P(Anorexie = Nu)
Da	0,5	0,5
Nu	0,1	0,9

$$w(s_1) = \frac{P(G_N) \cdot P(A_D) \cdot P(F_N|G_N, A_D) \cdot P(O_D|F_N) \cdot P(X_D|F_N)}{P(G_N) \cdot P(A_D) \cdot P(F_N|G_N, A_D)} =$$

$$P(O_D|F_N) \cdot P(X_D|F_N) = 0.2 \cdot 0.1 = 0.02$$

Sampling 2

- $G \sim M(Da : 0.1, Nu : 0.9) \Rightarrow G_D$
- $A \sim M(Da : 0.05, Nu : 0.95) \Rightarrow A_N$
- $F \sim M(Da : 0.7, Nu : 0.3) \Rightarrow F_D$
- O_D
- X_D
- $s_2 = (G_D, A_N, F_D, O_D, X_D)$

P(Gripă = Da)	P(Gripă = Nu)
0,1	0,9

P(Abces = Da)	P(Abces = Nu)
0,05	0,95

Gripă	Abces	P(Febră = Da)	P(Febră = Nu)
Da	Da	0,8	0,2
Da	Nu	0,7	0,3
Nu	Da	0,25	0,75
Nu	Nu	0,05	0,95

Febră	P(Oboseală = Da)	P(Oboseală = Nu)
Da	0,6	0,4
Nu	0,2	0,8

Febră	P(Anorexie = Da)	P(Anorexie = Nu)
Da	0,5	0,5
Nu	0,1	0,9

$$w(s_2) = \frac{P(G_D) \cdot P(A_N) \cdot P(F_D|G_D, A_N) \cdot P(O_D|F_D) \cdot P(X_D|F_D)}{P(G_D) \cdot P(A_N) \cdot P(F_D|G_D, A_N)} =$$

$$P(O_D|F_D) \cdot P(X_D|F_D) = 0.6 \cdot 0.5 = 0.3$$

Computing the final probabilities

$$P(G_D) = \frac{w(s_2)}{w(s_1) + w(s_2)} = \frac{0.3}{0.02 + 0.3} \approx 0.94$$

$$P(G_N) = \frac{w(s_1)}{w(s_1) + w(s_2)} = \frac{0.02}{0.02 + 0.3} \approx 0.06$$

- In this example, only two samples were used, so it is normal for the results to be different from the correct values: (0.4, 0.6)

Pseudocode

function LIKELIHOOD-WEIGHTING(X, $\mathbf{e}$, bn, N) **returns** an estimate of $\mathbf{P}(X|\mathbf{e})$
  inputs: X, the query variable
    $\mathbf{e}$, observed values for variables $\mathbf{E}$
    bn, a Bayesian network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$
    N, the total number of samples to be generated
  local variables: $\mathbf{W}$, a vector of weighted counts for each value of X, initially zero

  for j = 1 to N **do**
    $\mathbf{x}$, $w \leftarrow$ WEIGHTED-SAMPLE(bn, $\mathbf{e}$)
    $\mathbf{W}[x] \leftarrow \mathbf{W}[x] + w$ where x is the value of X in $\mathbf{x}$
  return NORMALIZE($\mathbf{W}$)

---

function WEIGHTED-SAMPLE(bn, $\mathbf{e}$) **returns** an event and a weight

  $w \leftarrow 1$; $\mathbf{x} \leftarrow$ an event with n elements initialized from $\mathbf{e}$
  foreach variable X_i **in** $X_1, \ldots, X_n$ **do**
    if X_i is an evidence variable with value x_i in $\mathbf{e}$
      then $w \leftarrow w \times P(X_i = x_i \mid parents(X_i))$
      else $\mathbf{x}[i] \leftarrow$ a random sample from $\mathbf{P}(X_i \mid parents(X_i))$
  return $\mathbf{x}$, w

Discussion

- The accuracy of the probability estimation increases with the number of network samples
- The algorithm does not converge well when there are many evidence nodes

Bayesian network applications

- Systems for medical diagnosis, for example, diagnosis of liver disease
- Predicting defects and reliability of software products, for example, *AgenaRisk*, used by Motorola, Philips and Siemens
- *TrueSkill* (Microsoft Research): player ranking system based on the team’s final results

Bayesian Networks

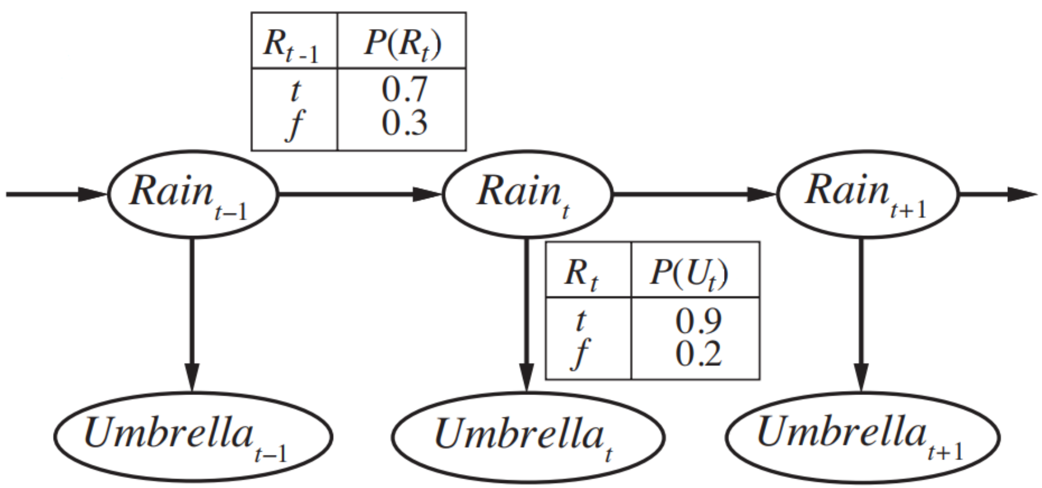
1. Probabilities
2. Bayesian Networks
3. Exact and Approximate Inferences
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

Dynamic Bayesian network

- A dynamic Bayesian network is a Bayesian network that represents a temporal probability model
- It is organized into time slices
- Each partition can have several state variables $\mathbf{X}_t$ and evidence variables $\mathbf{E}_t$
- To build an DBN, one must specify:
  - The prior distribution of state variables $P(\mathbf{X}_0)$
  - The transition model $P(\mathbf{X}_{t+1} \mid \mathbf{X}_t)$
  - The observation or sensor model $P(\mathbf{E}_t \mid \mathbf{X}_t)$

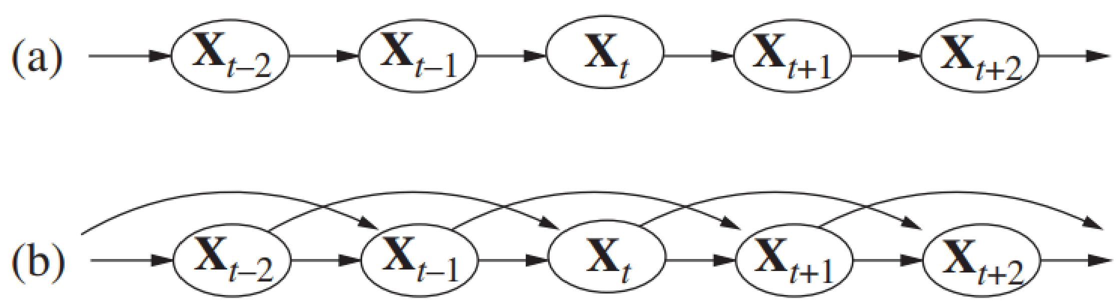
Example

Consider the following example: You are the security guard stationed at a secret underground installation. You want to know whether it's raining today, but your only access to the outside world occurs each morning when you see the director coming in with, or without, an umbrella. For each day t, the set $\mathbf{E}_t$ thus contains a single evidence variable $Umbrella_t$ or U_t for short (whether the umbrella appears), and the set $\mathbf{X}_t$ contains a single state variable $Rain_t$ or R_t for short (whether it is raining).



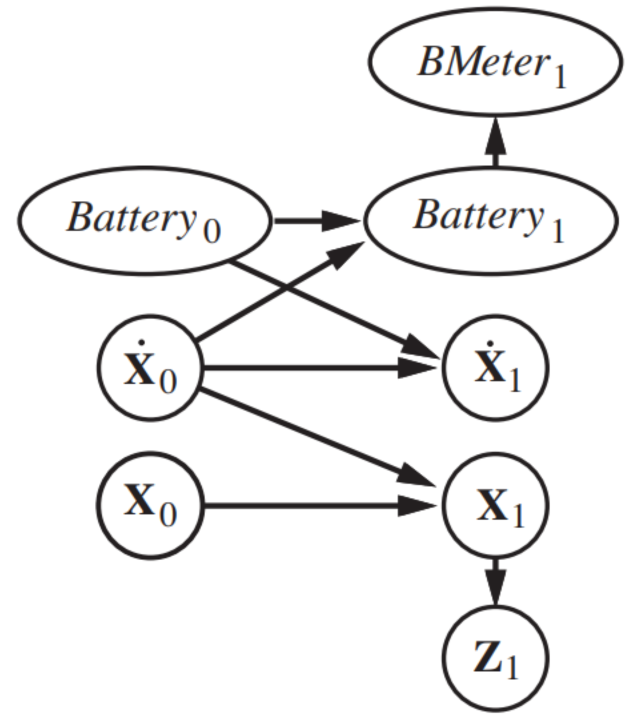
Modeling Markov processes

- DBN can model Markov processes
- The Markov assumption: the current state depends on a fixed finite number of previous states
- The figure shows:
  - a) a first order Markov process
  - b) a second order Markov process



Modeling arbitrary transitions

- Variables that reduce the problem to a first-order Markov process can be added
- Example: a robot with position **x**, speed $\dot{\mathbf{x}}$ and battery level *Battery* that depends on all previous maneuvers. An observation variable (sensor) for the *BMeter* battery is added. **z** is the observed variable for position

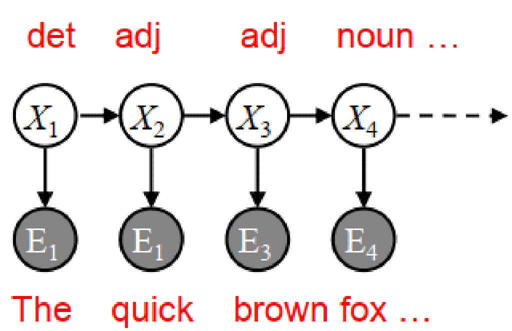


Other assumptions

- The changes in the environment are caused by a stationary process (≠ static)
  - The transition model $P(\mathbf{X}_{t+1} \mid \mathbf{X}_t)$ is the same for any t
- The sensors follow a first-order Markov process
  - $P(\mathbf{E}_t \mid \mathbf{X}_{0:t}, \mathbf{E}_{0:t\text{-}1}) = P(\mathbf{E}_t \mid \mathbf{X}_t)$
  - Current observations only depend on the state of the current moment

DBN applications

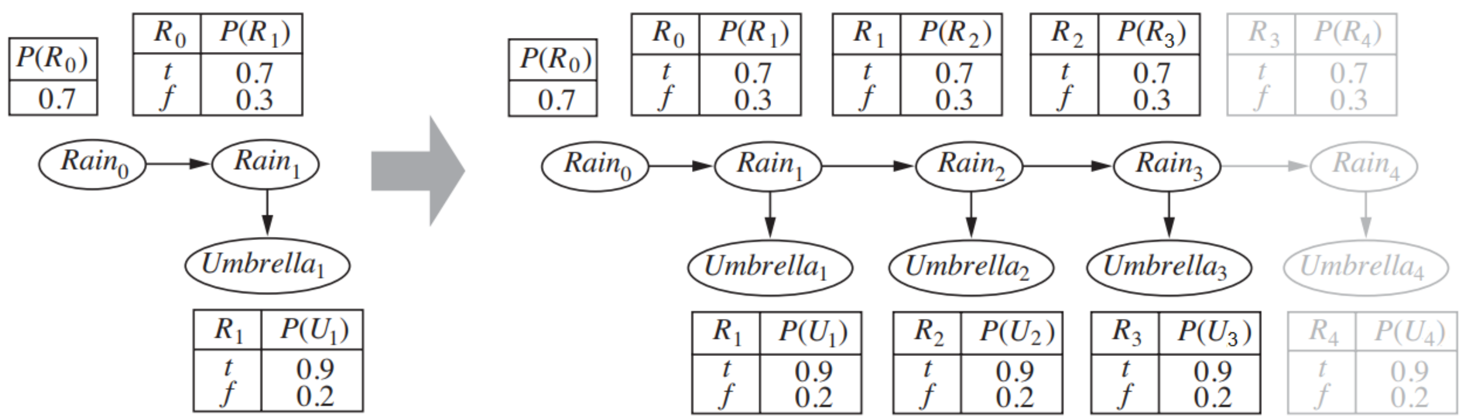
- Identification of the parts of speech
  - Observations: words (possibly thousands)
  - States: the labels of the speech parts



- Automatic translation
  - Observations: words
  - States: translating options
- Tracking the positions of robots or self-driving cars
  - Observations: location information from sensors
  - States: positions on the map

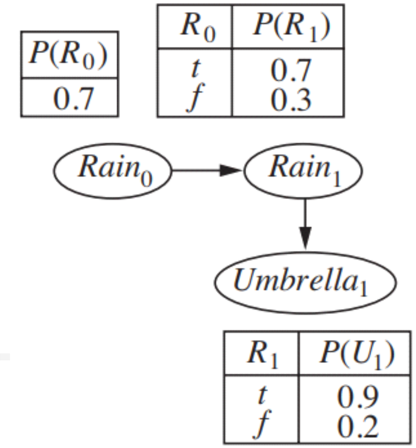
Inference in DBNs

- An DBN can be transformed into a simple BN by unrolling

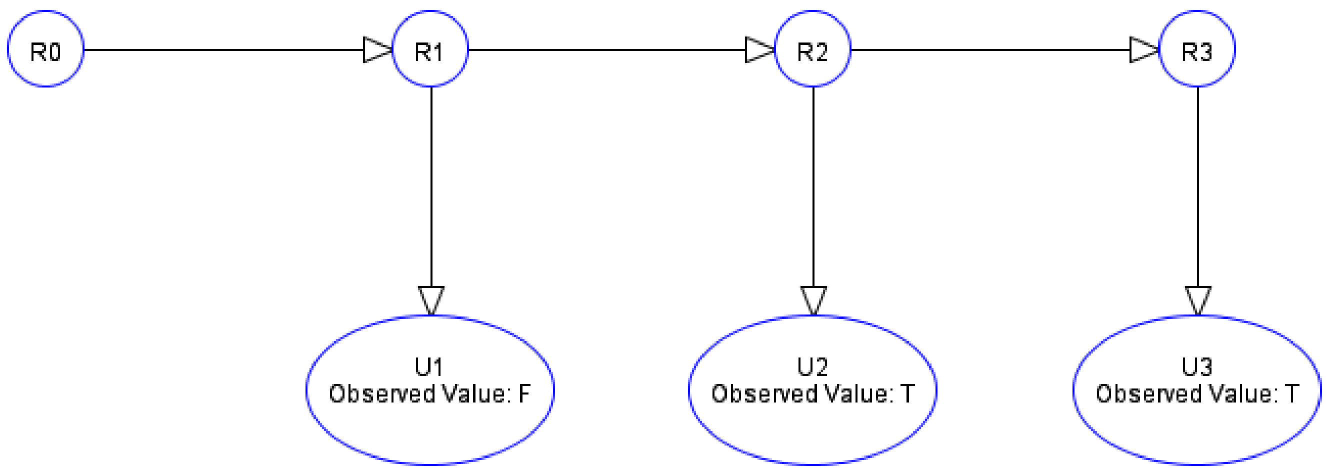


- Simple BN inference algorithms could be applied

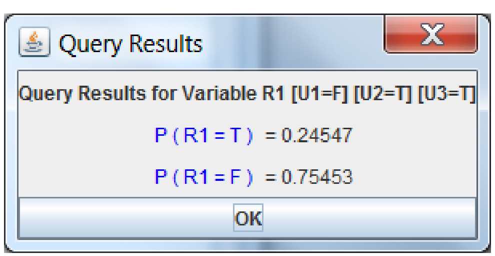
Example



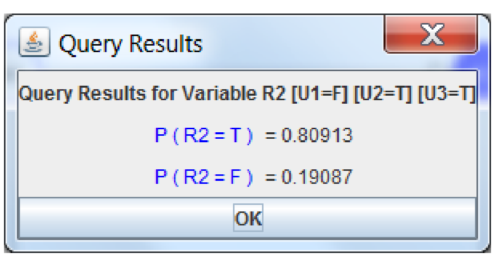
$P(R_0)$
0.7
R_0 | $P(R_1)$
t 0.7
f 0.3
$Rain_0$
$Rain_1$
$Umbrella_1$
R_1 | $P(U_1)$
t 0.9
f 0.2



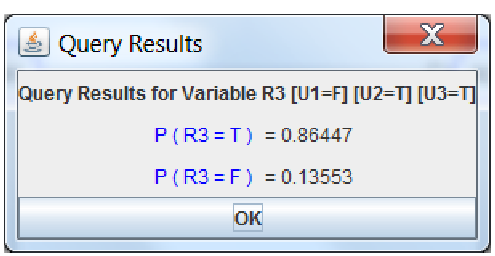
R0
R1
R2
R3
U1
Observed Value: F
U2
Observed Value: T
U3
Observed Value: T



Query Results
Query Results for Variable R1 [U1=F] [U2=T] [U3=T]
P (R1 = T) = 0.24547
P (R1 = F) = 0.75453
OK



Query Results
Query Results for Variable R2 [U1=F] [U2=T] [U3=T]
P (R2 = T) = 0.80913
P (R2 = F) = 0.19087
OK



Query Results
Query Results for Variable R3 [U1=F] [U2=T] [U3=T]
P (R3 = T) = 0.86447
P (R3 = F) = 0.13553
OK

Bayesian Networks

1. Probabilities
2. Bayesian Networks
3. Exact and Approximate Inferences
4. Dynamic Bayesian Networks
5. Particle Filtering
6. Conclusions

Particle filtering

- The complexity of the exact inference algorithms specific to simple BNs for unrolled DBNs is exponential
- There are other exact and approximate inference methods specific to DBN
- For example, the approximate method of particle filtering
- Filtering means the determination of $P(\mathbf{X}_t \mid \mathbf{E}_{0:t})$ at the current time t

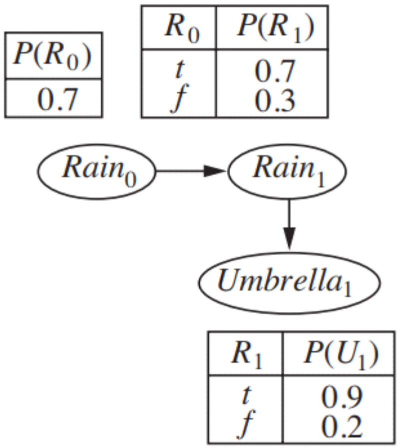
Particle filtering

- First, a population of *n* particles is created with the sampled state from the prior distribution $P(\mathbf{X}_0)$
- Then, repeat the following steps for each time step:
  - Each sample (particle state) is propagated forward by sampling the next state based on the transition model $P(\mathbf{X}_{t+1} \mid \mathbf{X}_t)$
  - Each particle is weighted by the probability assigned to the new evidence $P(\mathbf{E}_{t+1} \mid \mathbf{X}_{t+1})$
  - The population is re-sampled to generate a new population of *n* particles. Each new particle is selected from the current population. The probability of a particle being selected is proportional to its weight. The new particles have no weights

Example with rain and umbrella
Initialization

- $t = 0$

- $p_1 \sim M(T : 0.7, F : 0.3) \Rightarrow R_T$
- $p_2 \sim M(T : 0.7, F : 0.3) \Rightarrow R_F$
- $p_3 \sim M(T : 0.7, F : 0.3) \Rightarrow R_T$
- $p_4 \sim M(T : 0.7, F : 0.3) \Rightarrow R_F$

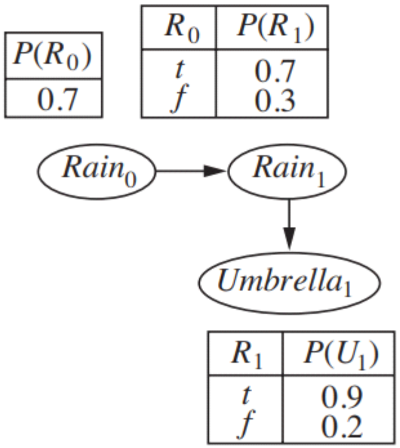


Phase 1. Propagation

- $t = 1$

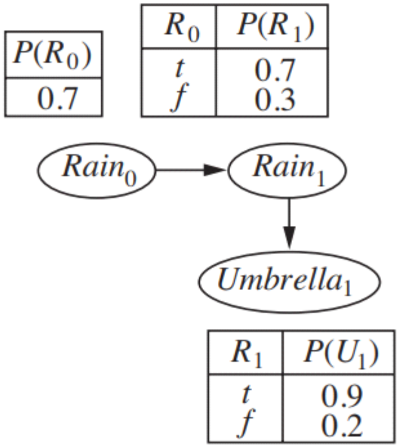
- $p_1 (R_T) \sim M(T : 0.7, F : 0.3) \Rightarrow R_T$
- $p_2 (R_F) \sim M(T : 0.3, F : 0.7) \Rightarrow R_F$
- $p_3 (R_T) \sim M(T : 0.7, F : 0.3) \Rightarrow R_T$
- $p_4 (R_F) \sim M(T : 0.3, F : 0.7) \Rightarrow R_T$

- At time step $t = 1$, evidence U_F appears



Phase 2. Weighting

- $w_1 = P(U_F \mid R_T) = 0.1$
- $w_2 = P(U_F \mid R_F) = 0.8$
- $w_3 = P(U_F \mid R_T) = 0.1$
- $w_4 = P(U_F \mid R_T) = 0.1$



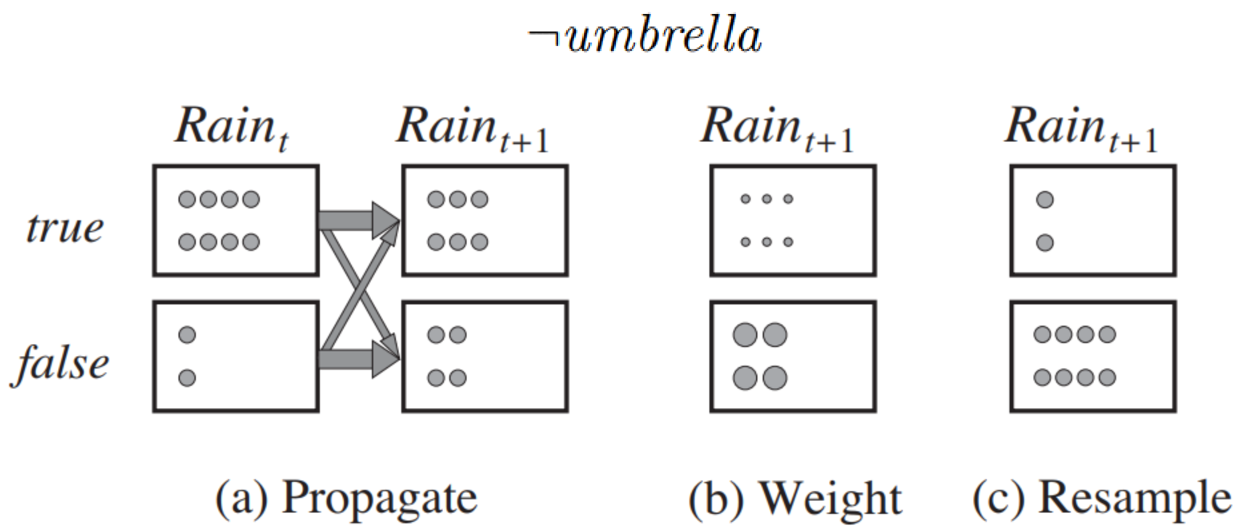
Phase 3. Resampling

- $p_i \sim M(T : 0.1, F : 0.8, T : 0.1, T : 0.1)$

- $p_1 \Rightarrow R_F$
- $p_2 \Rightarrow R_F$
- $p_3 \Rightarrow R_T$
- $p_4 \Rightarrow R_F$

- $\Rightarrow P(R_T) = 25\%, P(R_F) = 75\%$

Example with rain and umbrella



The particle filtering update cycle for the umbrella DBN with $N = 10$, showing the sample populations of each state. (a) At time t, 8 samples indicate $rain$ and 2 indicate $\neg rain$. Each is propagated forward by sampling the next state through the transition model. At time $t + 1$, 6 samples indicate $rain$ and 4 indicate $\neg rain$. (b) $\neg umbrella$ is observed at $t + 1$. Each sample is weighted by its likelihood for the observation, as indicated by the size of the circles. (c) A new set of 10 samples is generated by weighted random selection from the current set, resulting in 2 samples that indicate $rain$ and 8 that indicate $\neg rain$.

Pseudocode

function PARTICLE-FILTERING($\mathbf{e}$, N, dbn) **returns** a set of samples for the next time step
  inputs: $\mathbf{e}$, the new incoming evidence
    N, the number of samples to be maintained
    dbn, a DBN with prior $\mathbf{P}(\mathbf{X}_0)$, transition model $\mathbf{P}(\mathbf{X}_1|\mathbf{X}_0)$, sensor model $\mathbf{P}(\mathbf{E}_1|\mathbf{X}_1)$
  persistent: S, a vector of samples of size N, initially generated from $\mathbf{P}(\mathbf{X}_0)$
  local variables: W, a vector of weights of size N

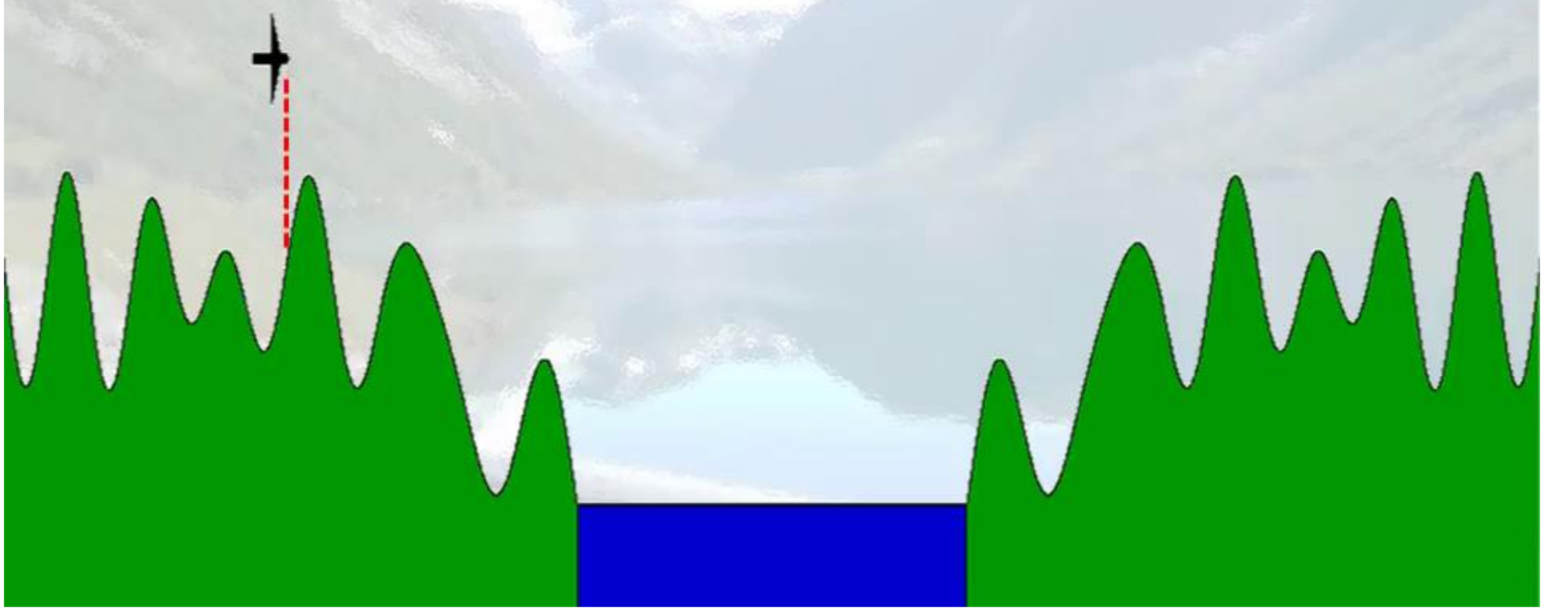
  for i = 1 to N **do**
    $S[i] \leftarrow$ sample from $\mathbf{P}(\mathbf{X}_1 \mid \mathbf{X}_0 = S[i])$ /* step 1 */
    $W[i] \leftarrow \mathbf{P}(\mathbf{e} \mid \mathbf{X}_1 = S[i])$ /* step 2 */
  $S \leftarrow$ WEIGHTED-SAMPLE-WITH-REPLACEMENT(N, S, W) /* step 3 */
  return S

So, this is our aicraft.

- We can measure the elevation (relative the sea)
- We can measure the distance to the ground
- We have the map

But we *don't* know where we are on the map!

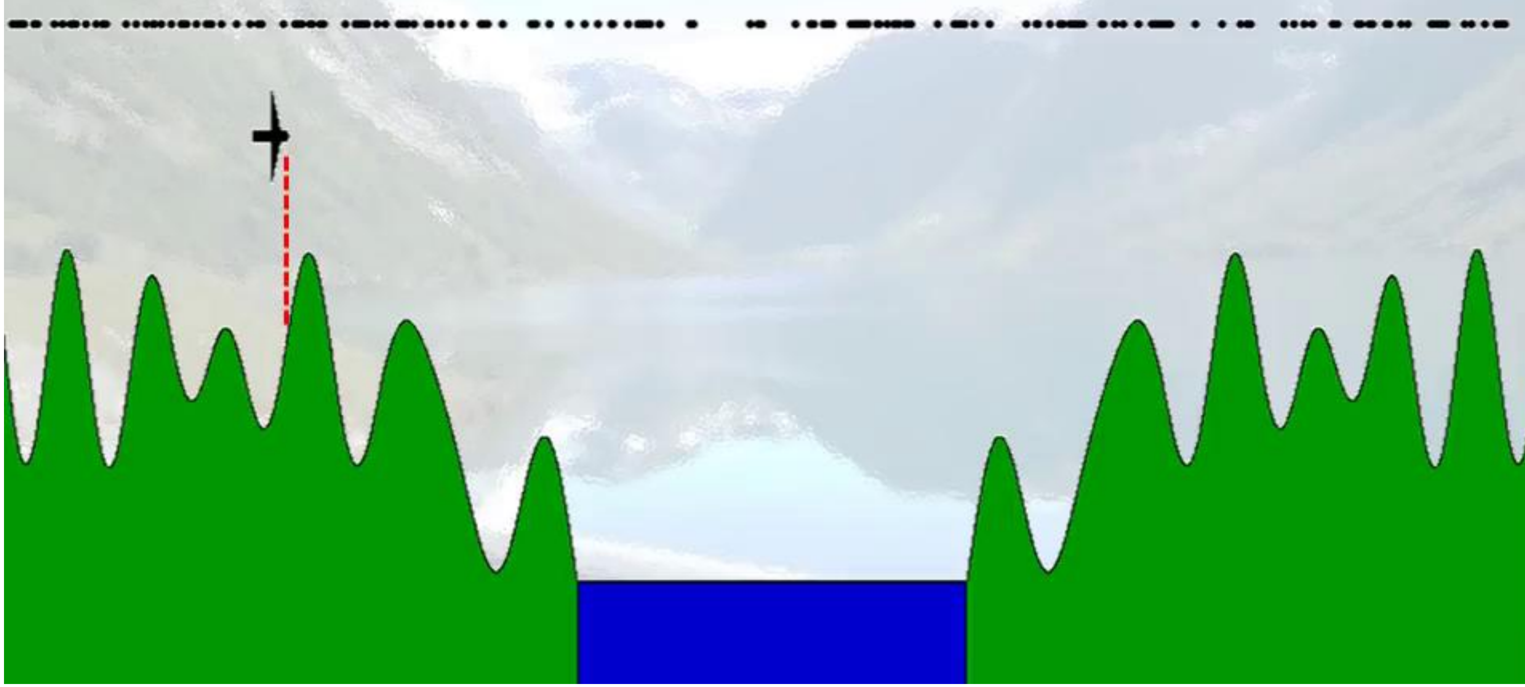
(Which could be useful to know, if it's dark or foggy…)



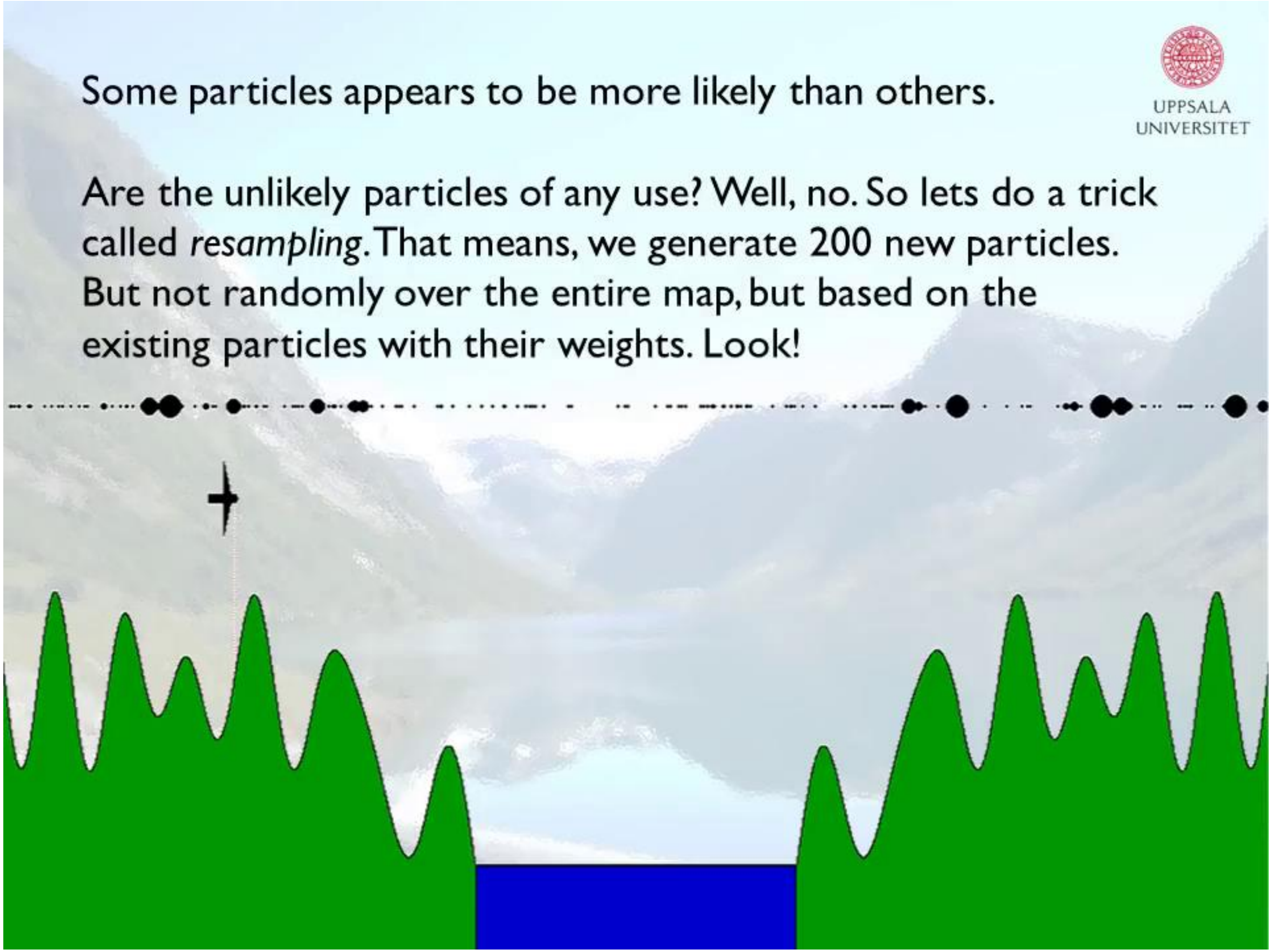
UPPSALA UNIVERSITET

https://www.youtube.com/watch?v=aUkBa1zMKv4

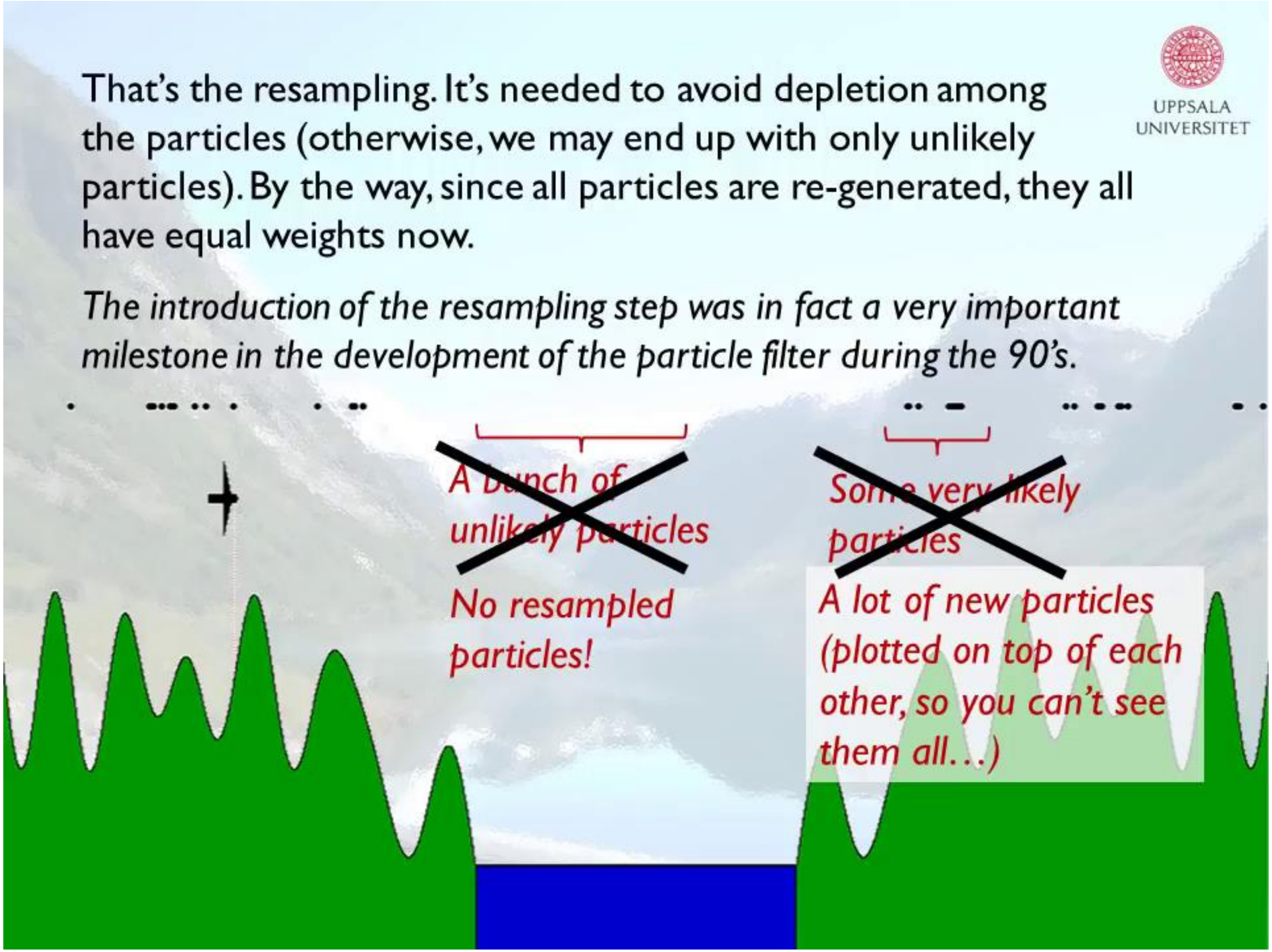
- We have measured the distance to the ground (with some uncertainty)
- We have the map
- We know how high we are flying (again with some uncertainty from the measurement device)



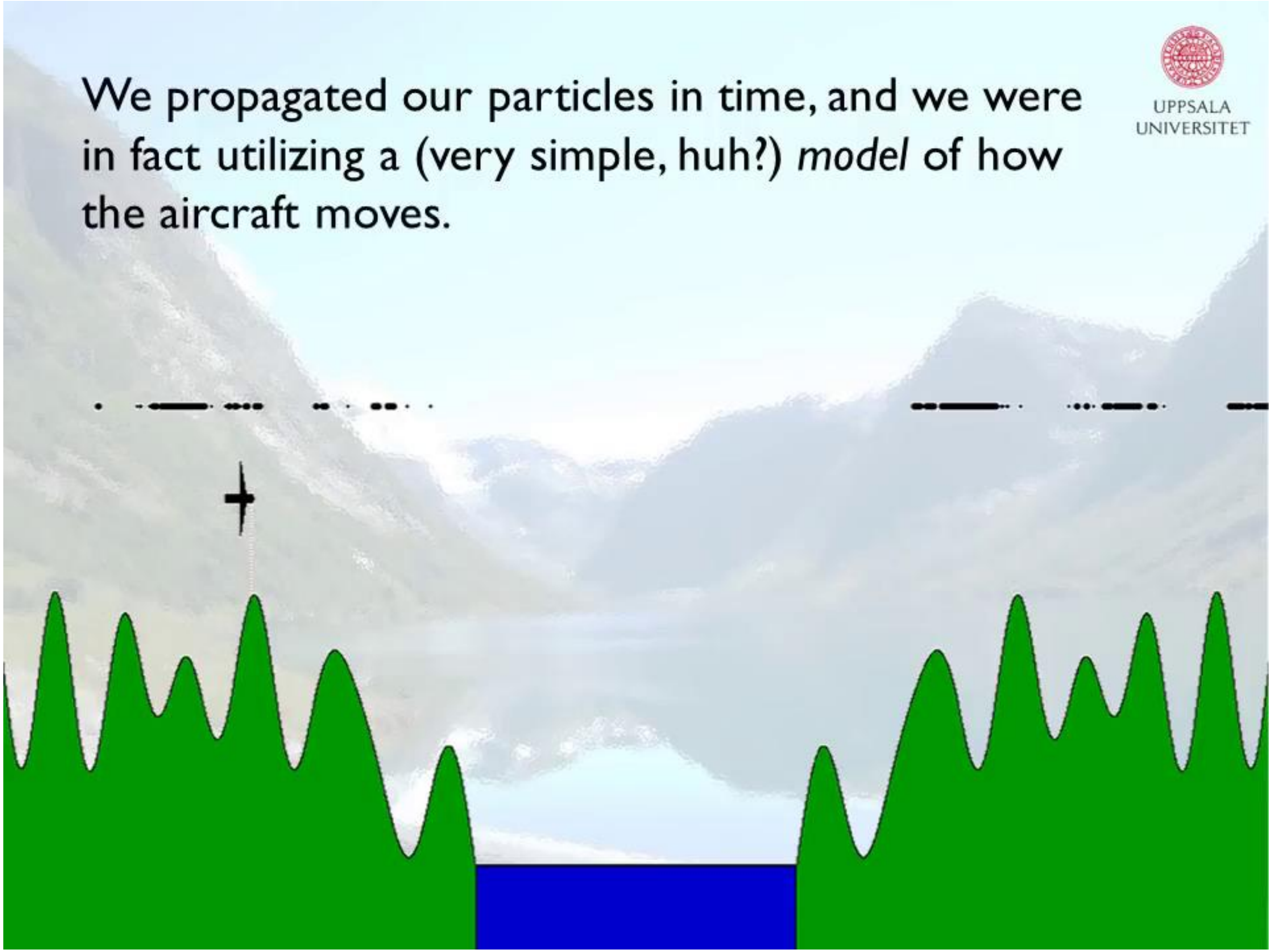
https://www.youtube.com/watch?v=aUkBa1zMKv4



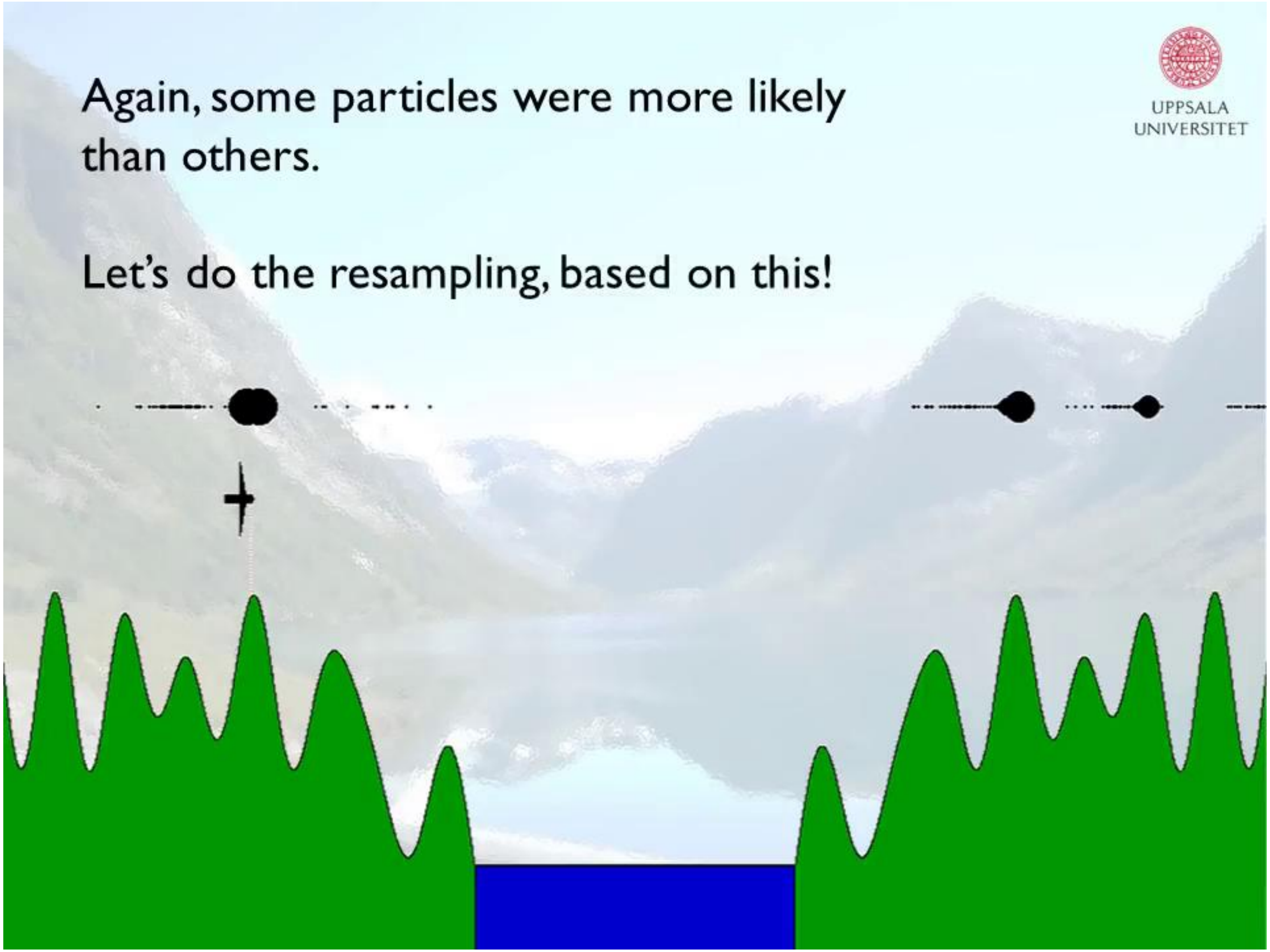
https://www.youtube.com/watch?v=aUkBa1zMKv4



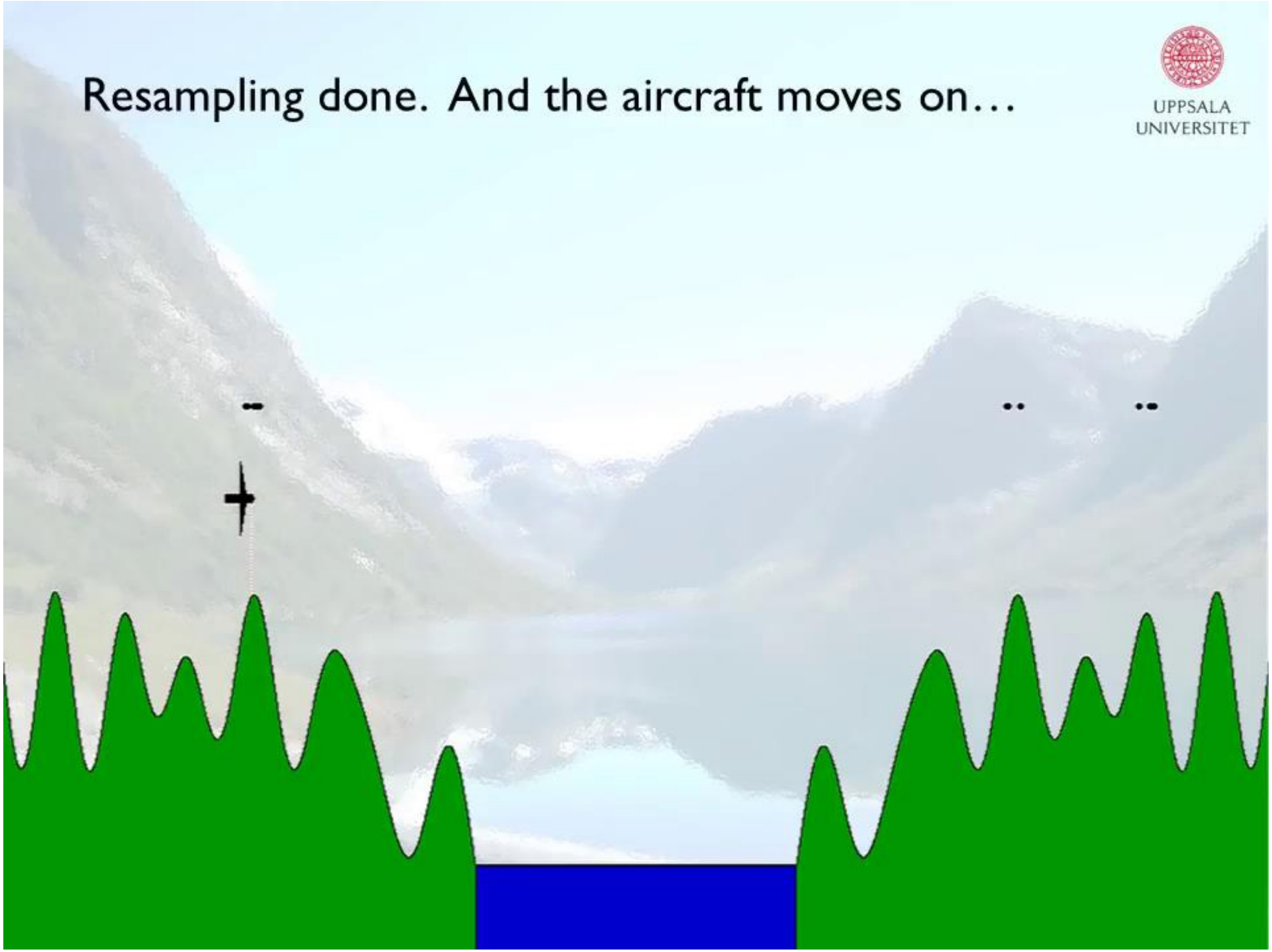
https://www.youtube.com/watch?v=aUkBa1zMKv4



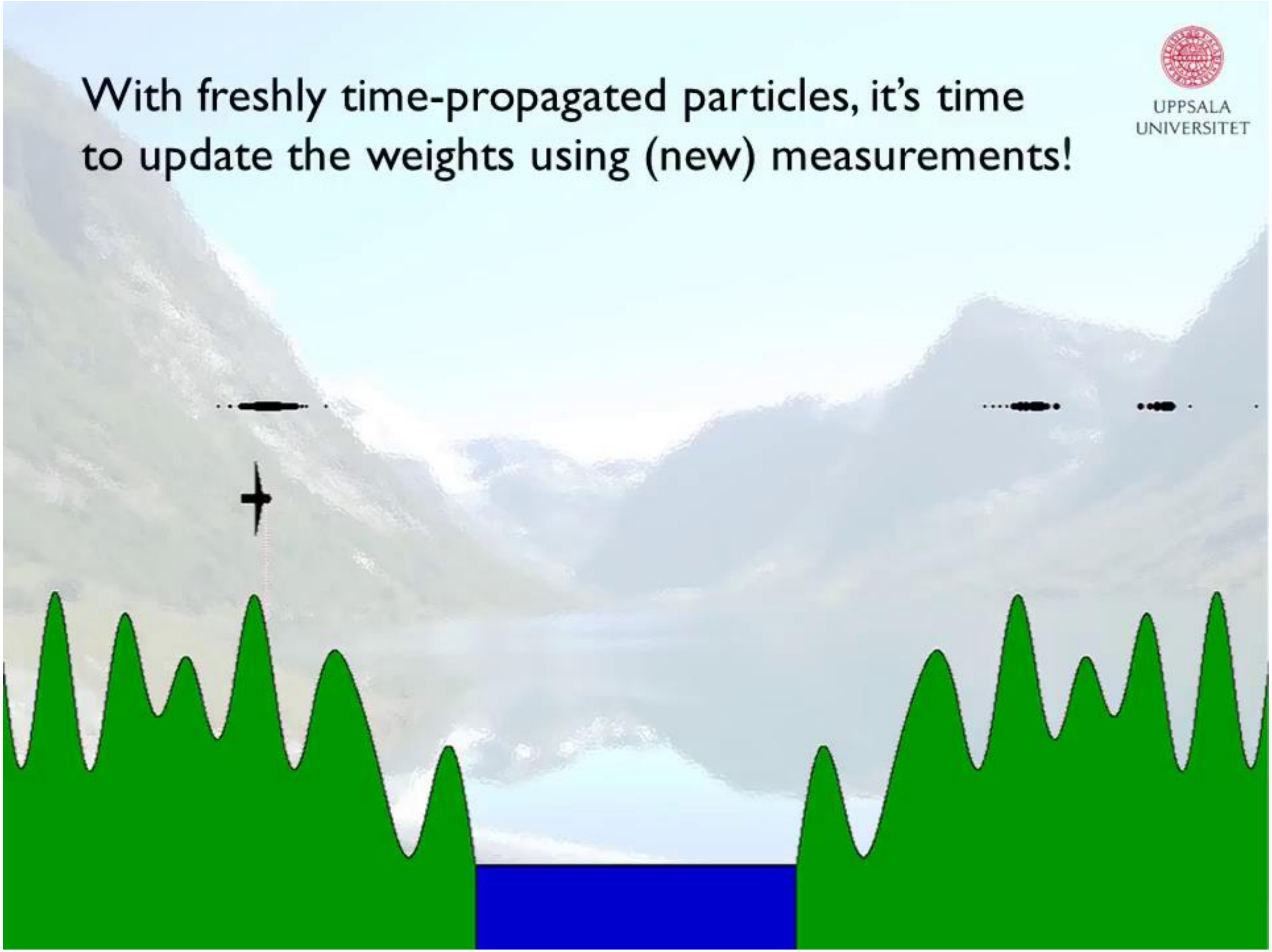
https://www.youtube.com/watch?v=aUkBa1zMKv4



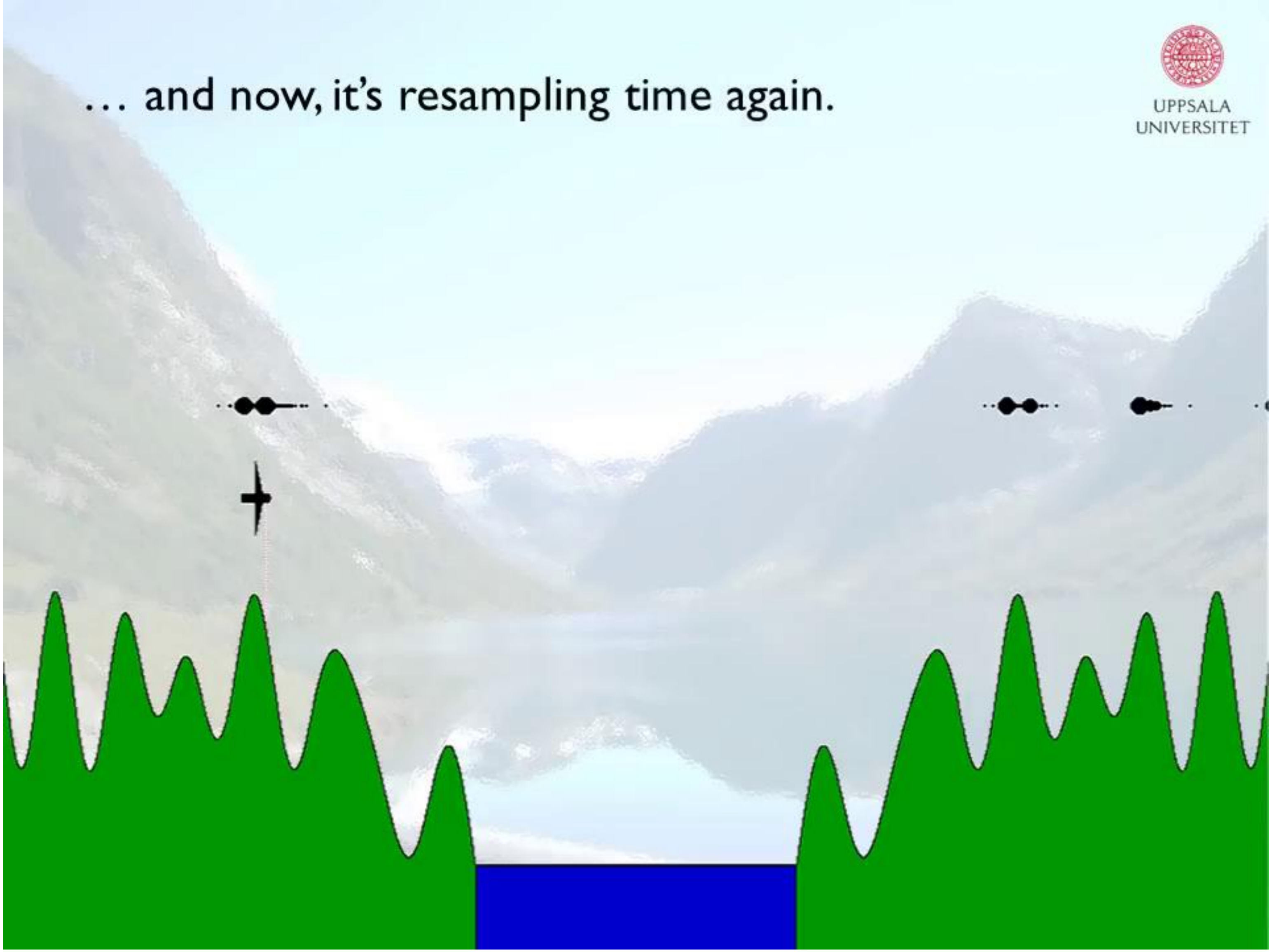
https://www.youtube.com/watch?v=aUkBa1zMKv4



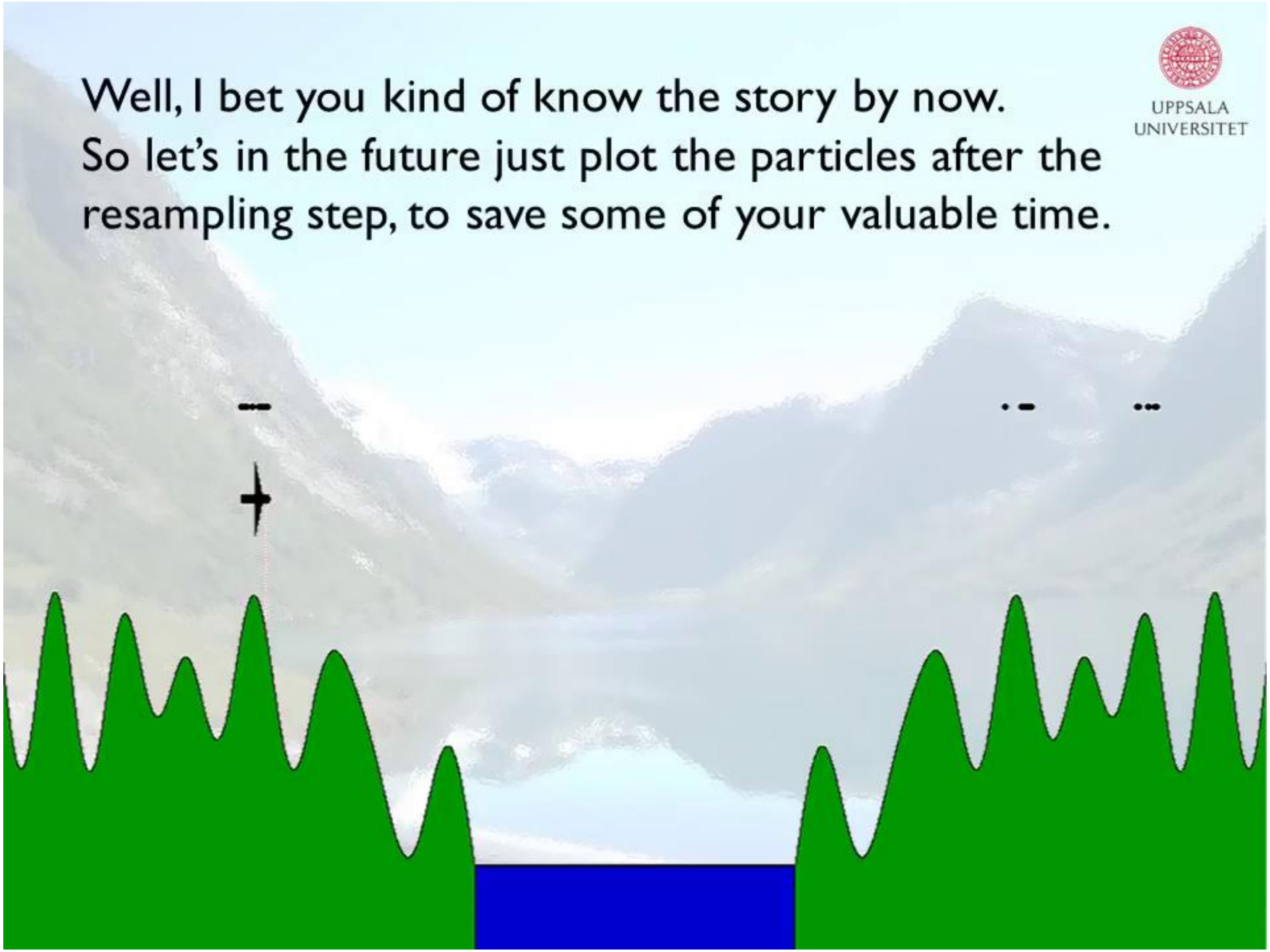
https://www.youtube.com/watch?v=aUkBa1zMKv4



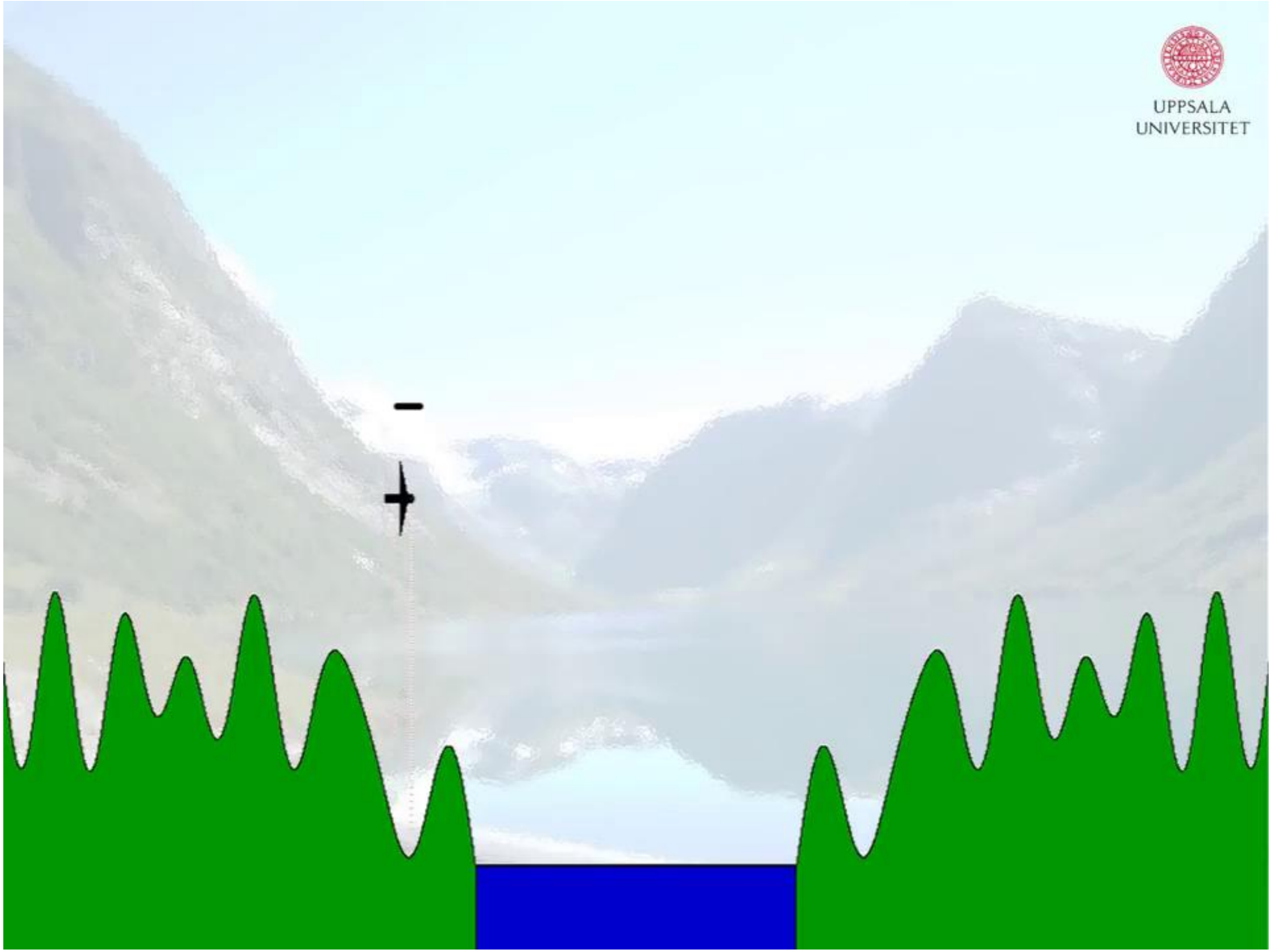
https://www.youtube.com/watch?v=aUkBa1zMKv4



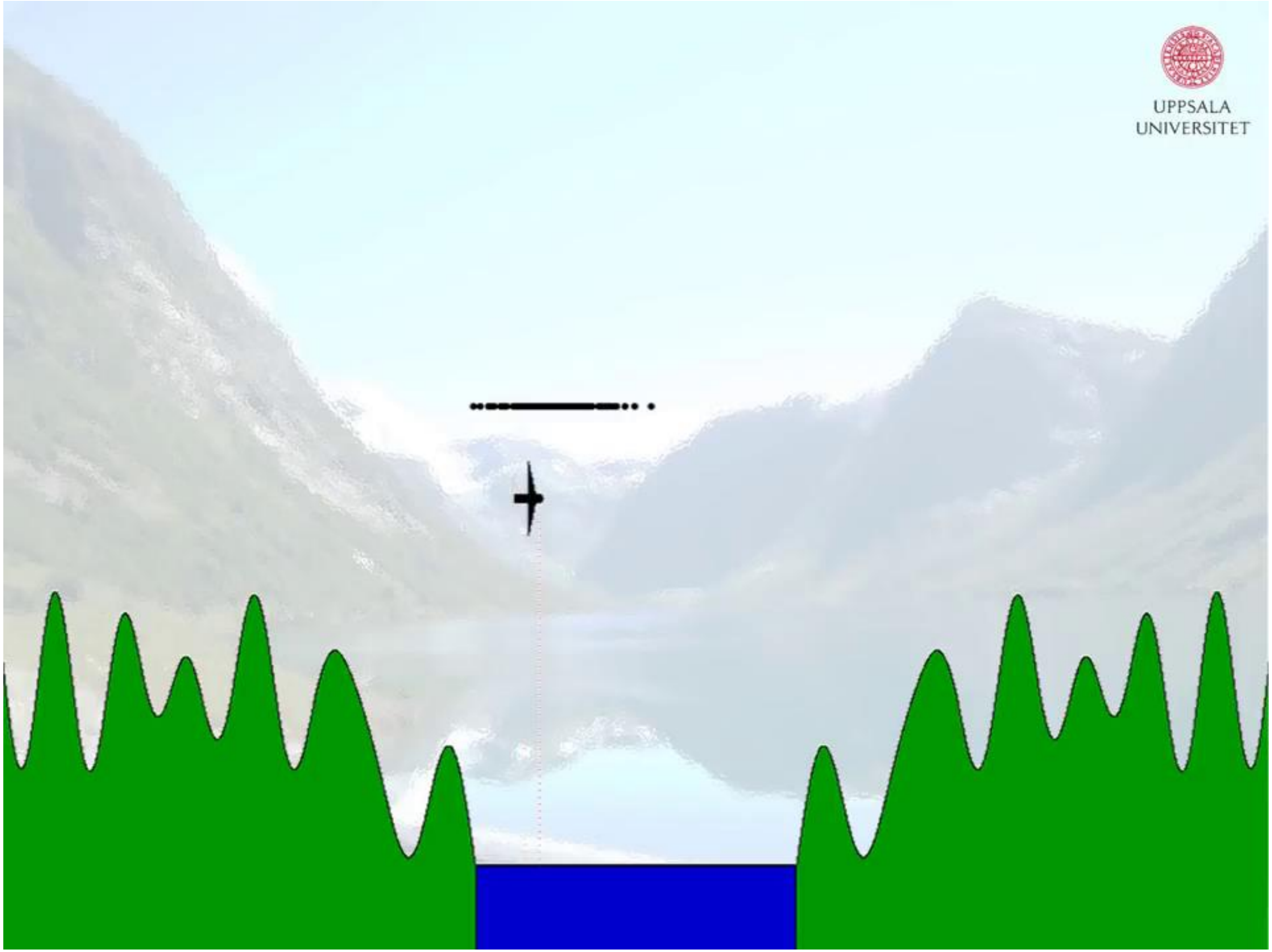
https://www.youtube.com/watch?v=aUkBa1zMKv4



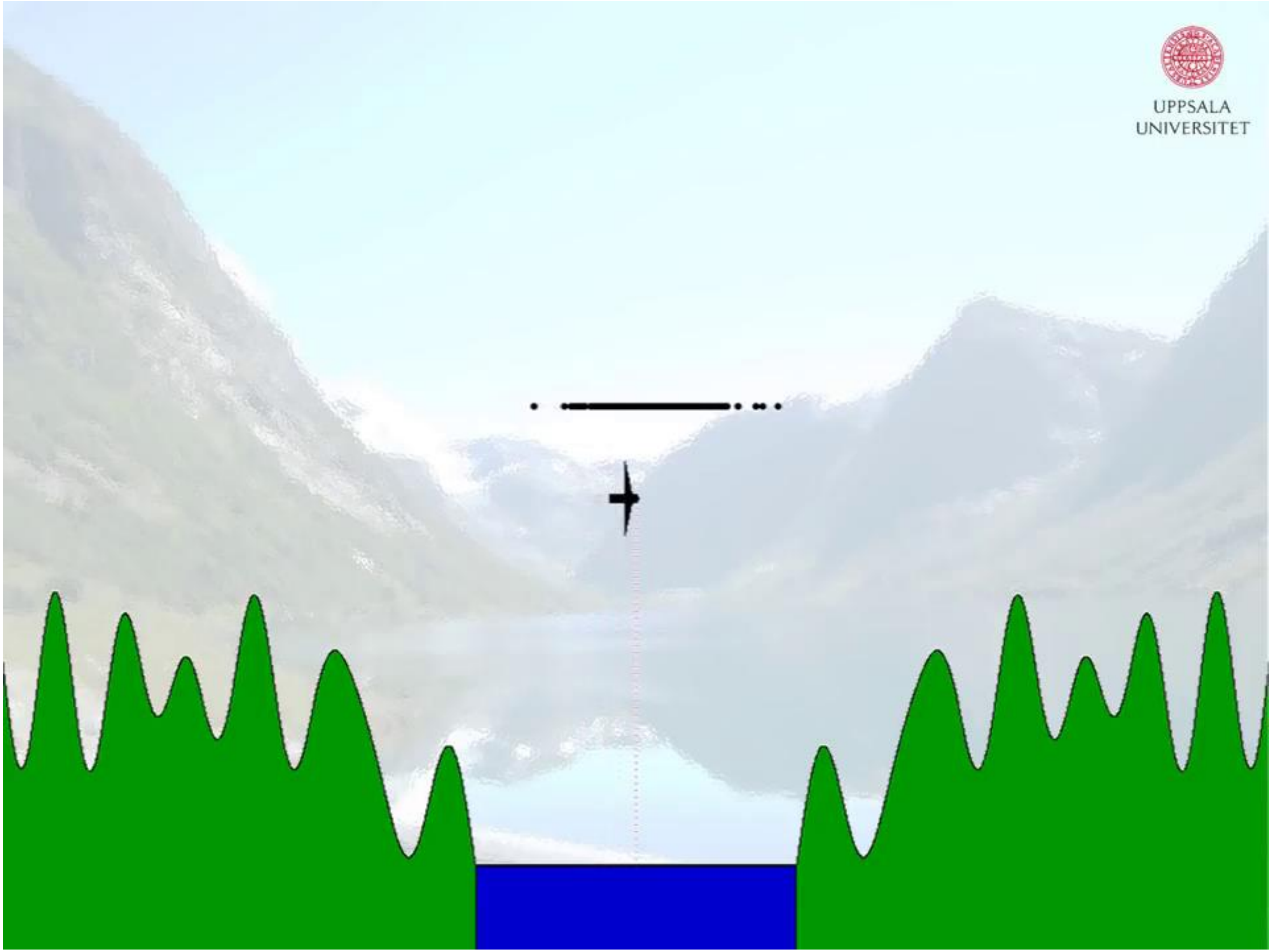
https://www.youtube.com/watch?v=aUkBa1zMKv4



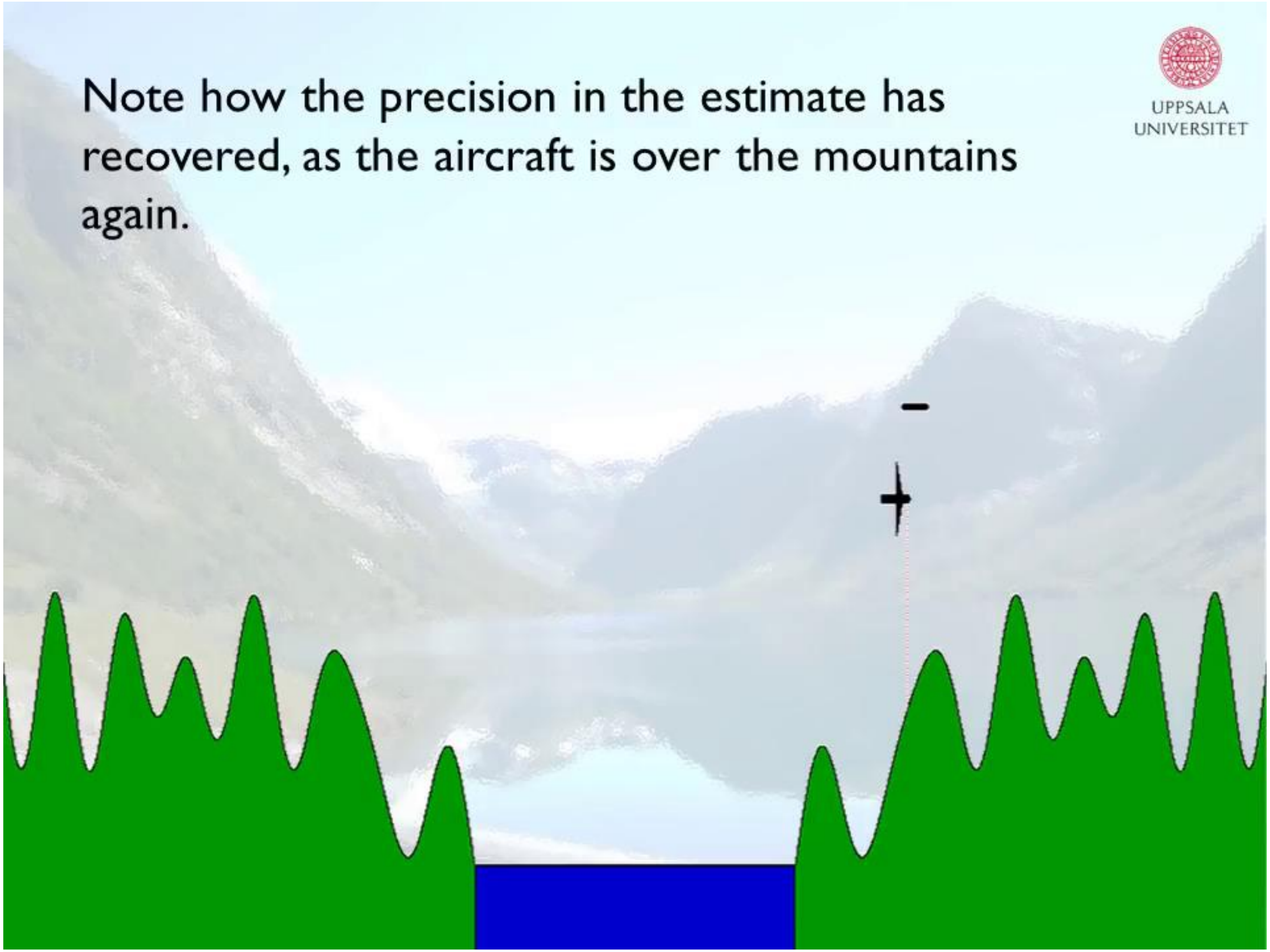
https://www.youtube.com/watch?v=aUkBa1zMKv4



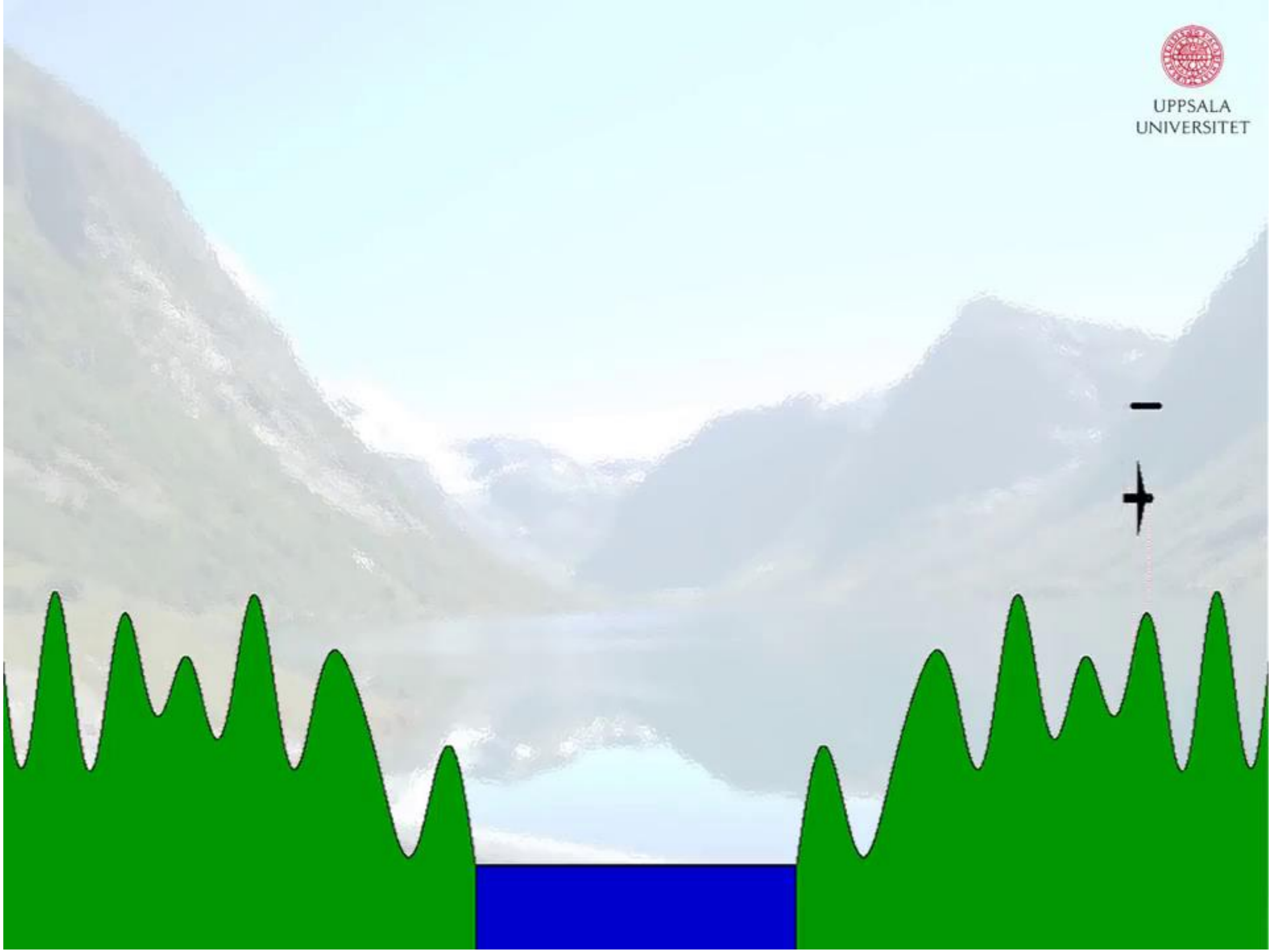
https://www.youtube.com/watch?v=aUkBa1zMKv4



https://www.youtube.com/watch?v=aUkBa1zMKv4



https://www.youtube.com/watch?v=aUkBa1zMKv4



https://www.youtube.com/watch?v=aUkBa1zMKv4

Advantages

- Constant time complexity
- It has been observed empirically that the approximation error remains bounded over time (theoretical analysis is difficult)
- The larger the number of particles, the better the approximation of probabilities

Conclusions

- Bayesian networks provide a concise way of representing the conditional independence relations in a domain and making inferences
- Dynamic Bayesian networks are Bayesian networks that represent temporal probability models
- There are many exact or approximate inference methods for Bayesian networks