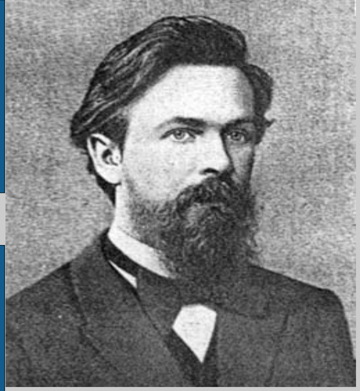
CSC421 Intro to Artificial Intelligence

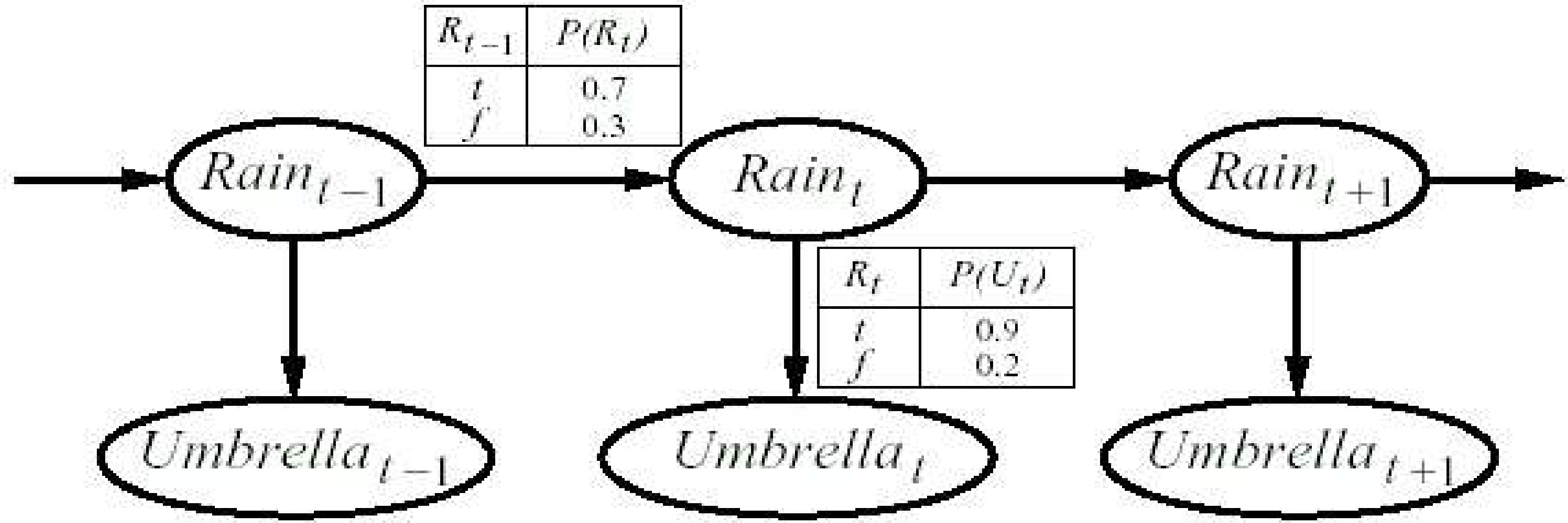
UNIT 27: Probabilistic Reasoning over Time



Markov Processes (or chains)

- Markov assumption
  - X_t depends on **bounded** subset of $X_{0:t-1}$
  - First-order: $P(X_t | X_{0:t-1}) = P(X_t | X_{t-1})$
  - Second-order: $P(X_t | X_{0:t-1}) = P(X_t | X_{t-2}, X_{t-1})$
  - ...
- Sensor Markov Assumption
  - $P(E_t | X_{0:t}, E_{0:t-1}) = P(E_t | X_t)$
- Stationary process
  - Transition model $P(X_t | X_{t-t})$ fixed for all t
  - Sensory model $P(E_t | X_t)$ fixed for all t

Example



First-order approximation not always true in real world

Possible fixes:

1) Increase order of Markov Process

2) Augment state e.g. add $Temp_t$, $Pressure_t$

Example : robot motion

Augment position and velocity with battery status

Inference Tasks

- Filtering: $P(X_t|e_{1:t})$
  - Belief state: input to decision process of agent
- Prediction: $P(X_{t+k}|e_{1:t})$ for k > 0
  - Evaluation of possible action sequences, like filter without full evidence
- Smoothing: $P(X_k|e_{1:t})$for 0 <= k <= t
  - Better estimate of past states, essential for learning
- Most likely explanation: $\text{argmax}_{x1:t} P(x_{1:t}|e_{1:t})$
  - Speech recognition, decoding over a noisy channel

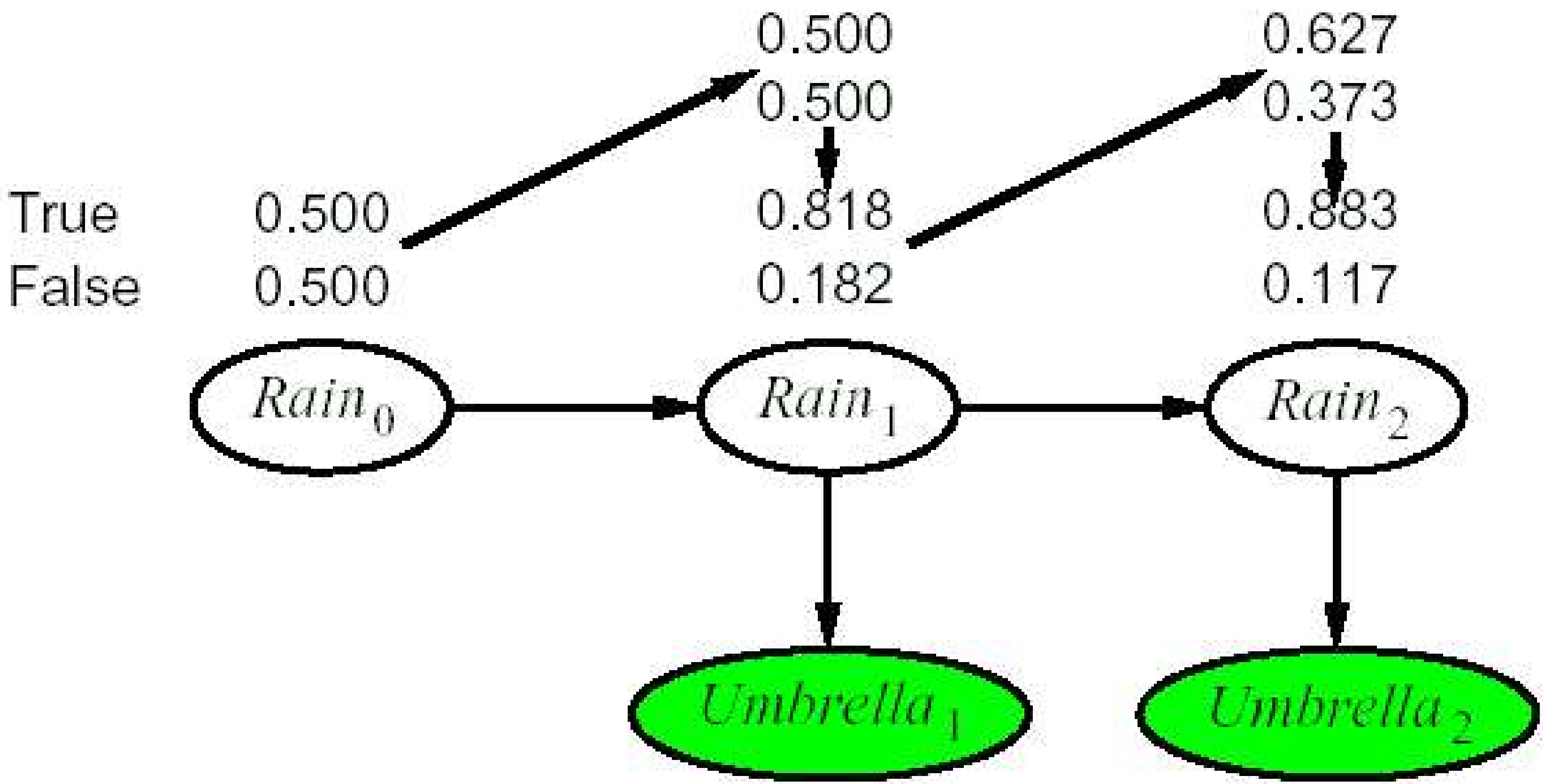
Filtering

- Aim: Device recursive state estimation:
  - $P(X_{t+1}|e_{1:t+1}) = f(e_{t+1},P(X_t|e_{1:t}))$
  - Current state projected forward from t to t+1, then updated using new evidence e_{t+1}
  - $P(X_{t+1}|e_{1:t+1}) = P(X_{t+1}|e_{1:t},e_{t+1})$
    $= a\,P(e_{t+1}|X_{t+1},e_{1:t})\,P(X_{t+1}|e_{1:t})$ (Bayes rule)
    $= a\,P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$ (Markov evidence)
  - Second term: one step prediction of next state
  - First term: updates this with new evidence – directly obtainable from sensor model

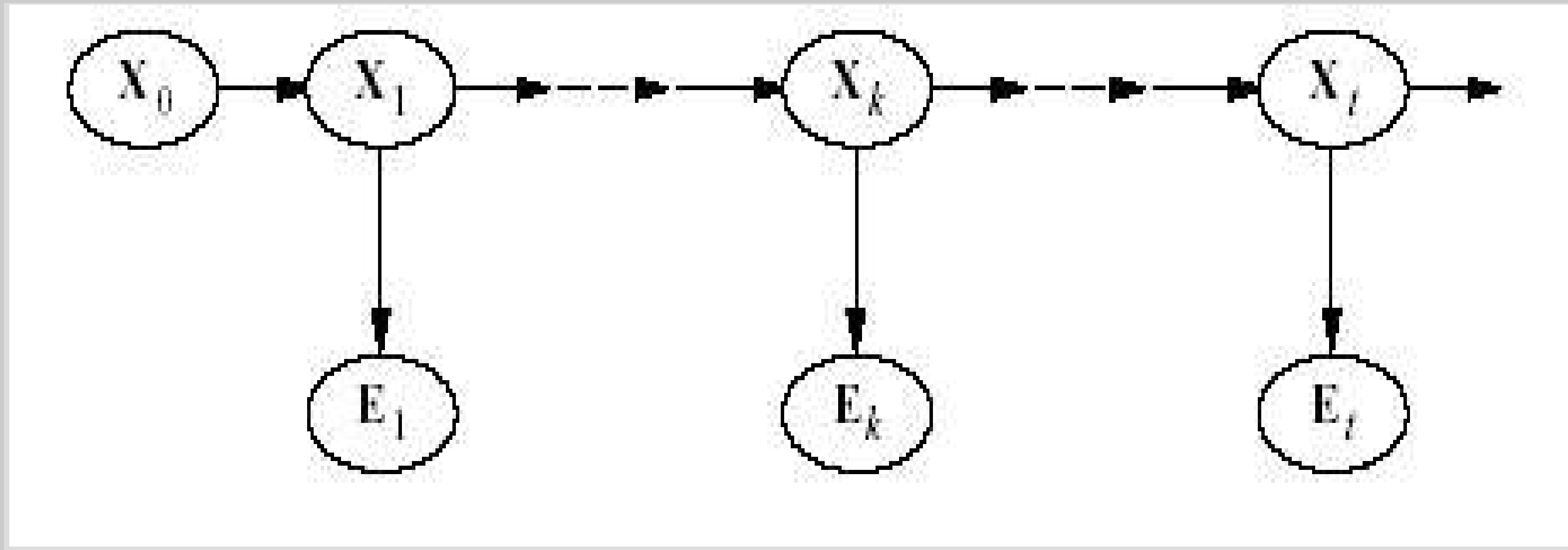
Filtering

- $P(X_{t+1}|e_{1:t+1}) = a\, P(e_{t+1}|X_{t+1})\, \sum_{xt} P(X_{t+1}|x_t, e_{1:t}) P(x_t|e_{1:t})$
  $= a\, P(e_{t+1}|X_{t+1})\, \sum_{xt} P(X_{t+1}|x_t) P(x_t|e_{1:t})$ (Markov)
- Within summation first factor is simply the transition model, second is the current state distribution - recursive
- Forward "message" propagation

Filtering Example



Smoothing



Process of computing the distribution over past states given evidence up to the present – In notation ?

Smoothing

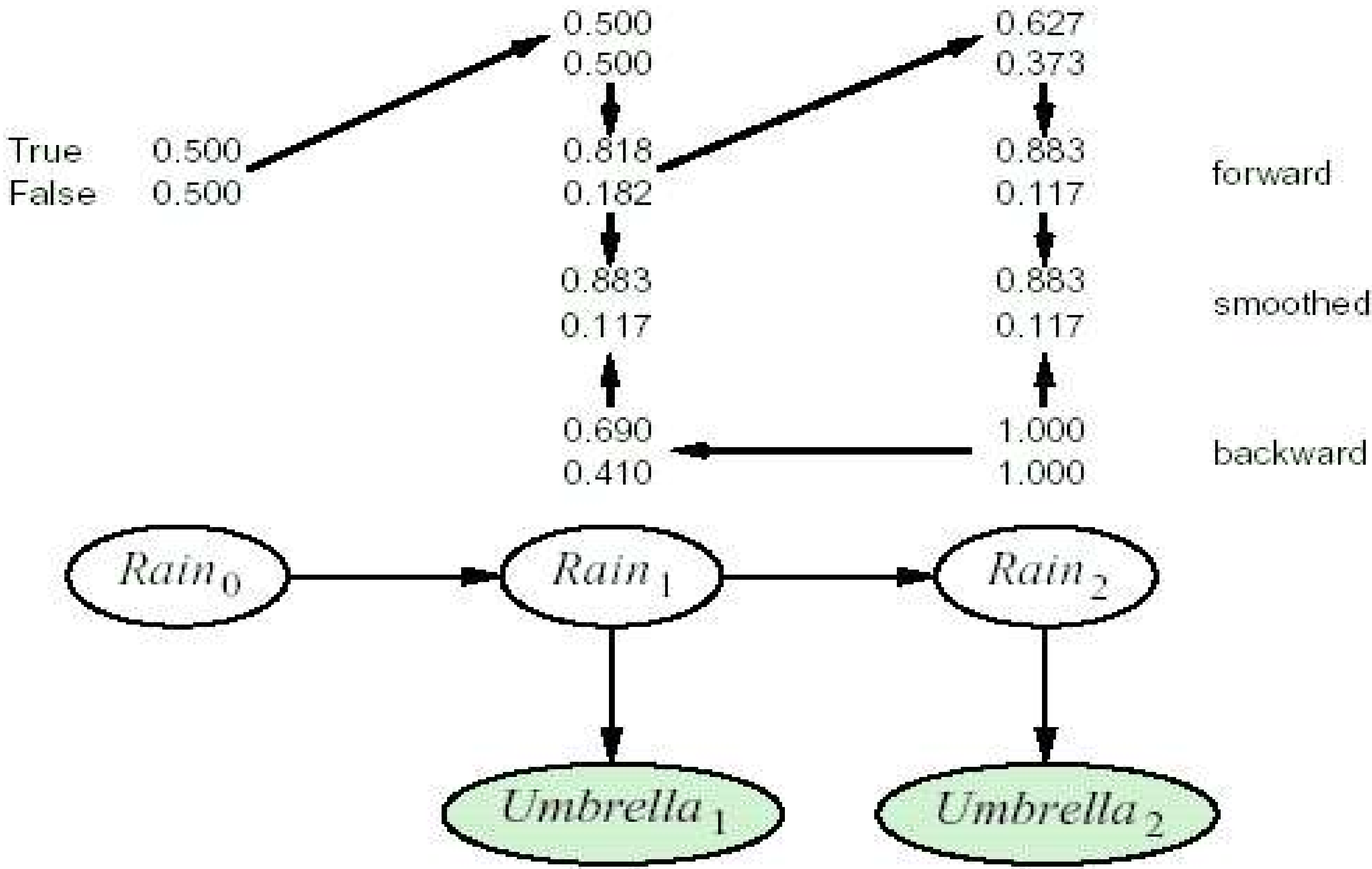
- Divide evidence $e_{1:t}$ to $e_{1:k}$, $e_{k+1,t}$
  - $P(X_k \mid e_{1:t}) = P(X_k \mid e_{1:k}, e_{k+1,t})$

    $= a\, P(X_k \mid e_{1:k})\, P(e_{k+1,t} \mid X_{k,}\, e_{k+1,t})$

    $= a\, P(X_k \mid e_{1:k})\, P(e_{k+1,t} \mid X_k)$

    $= a f_{1:k} b_{k+1:t}$
- Backward message computed by a backwards recursion

Smoothing example



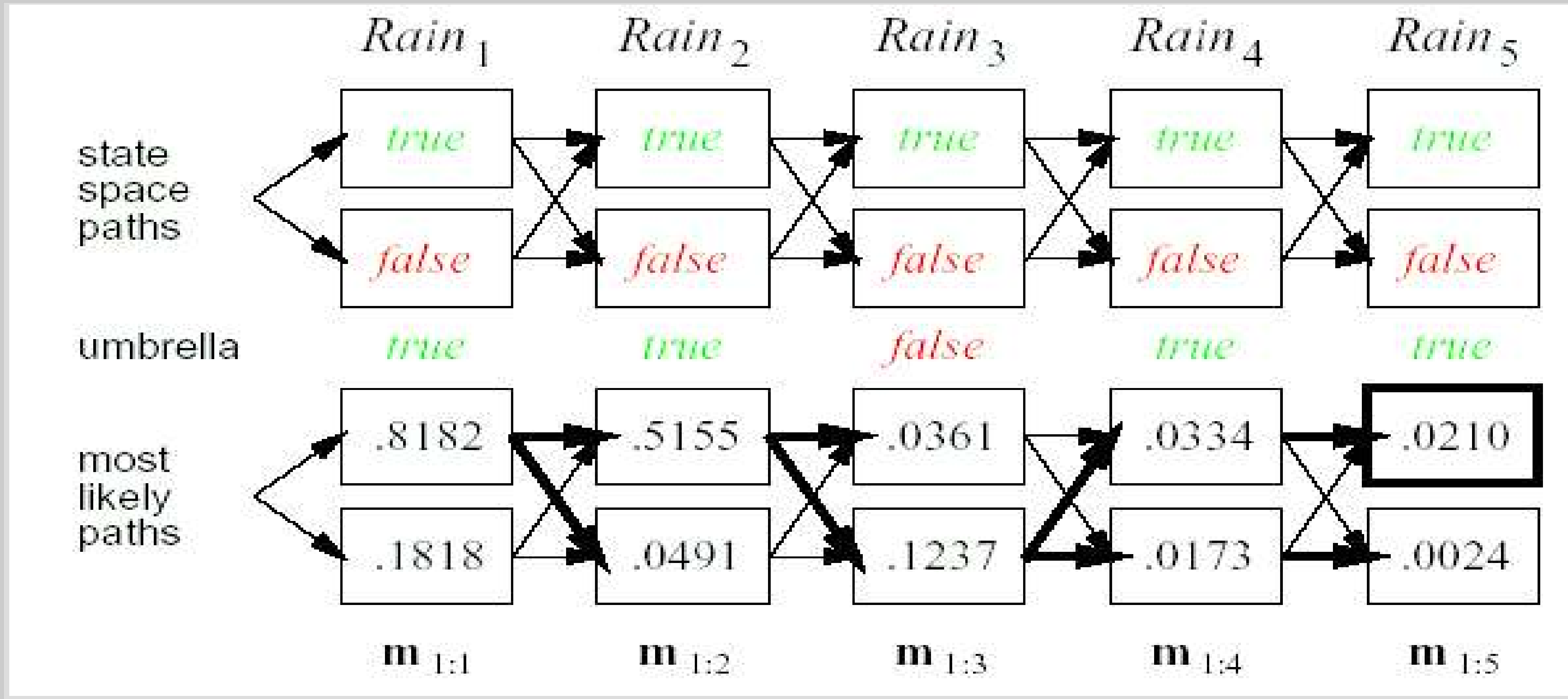
Forward-Backward Algorithm

- Suppose we want to “smooth” entire sequence
- Naïve Run smoothing once for every time step $O(t^2)$
- Dynamic programming
  - Cache results of forward filtering over entire sequence $O(t)$
- Forward-Backward algorithm
  - Backbone of computational methods from speech recognition to radar tracking of aircraft
- Modifications for online setting
  - Fixed-lag smoothing

Most likely explanation

- Most likely sequence ≠ Sequence of likely states
- Most likely path to each x_{t+1} = most likely path to some x_t plus one more step
- $\max_{x1...xt} P(x_1,...,x_t, X_{t+1} \mid e_{1:t+1})$ = $P(e_{t+1}|X_{t+1}) \max_{Xt}(P(X_{t+1} \mid x_t) \max_{x1...xt-1} P(x_1,...,x_{t-1}, x_t \mid e_{1:t}))$
- Similar to filtering sum is replaced by max
  - $m_{1:t} = \max_{x1...xt-1} P(x_1,...,x_{t-1}, X_t \mid e_{1:t})$
  - i.e $m_{1:t}(t)$ gives the probability of the most likely path to state i
  - Viterbi algorithm

Example



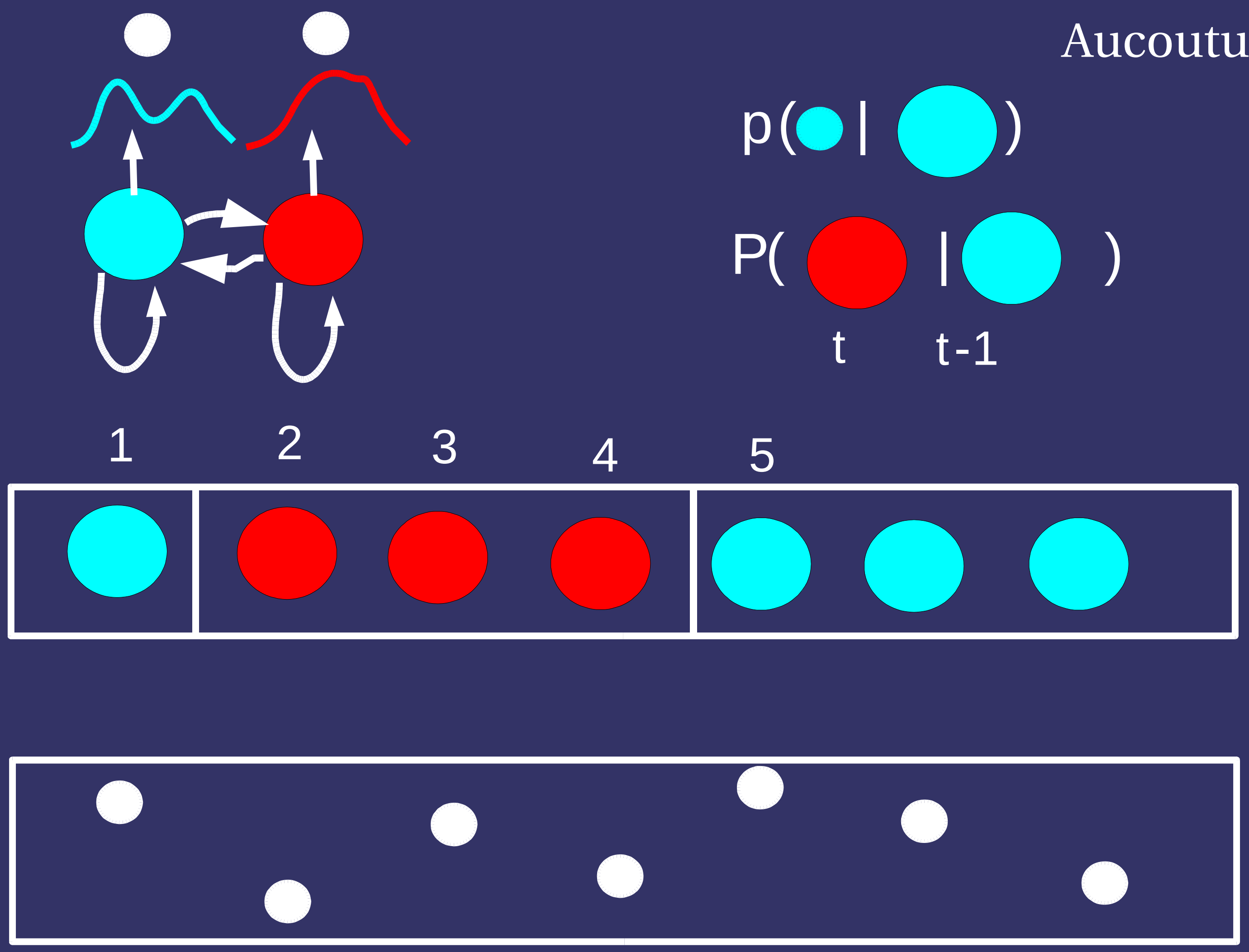
Viterbi: $m_{1:t+1} = P(e_{t+1}|X_{t+1}) \max_{X_t} (P(X_{t+1}| x_t) m_{1:t})$

Hidden Markov Models

- Dominant method for automatic speech recognition
- Temporal probabilistic model in which the state of the process is described by a single discrete random variable
- Simple and elegant matrix implementation of all the basic algorithms

HMM segmentation

Aucouturier & Sandler, AES 01



Model

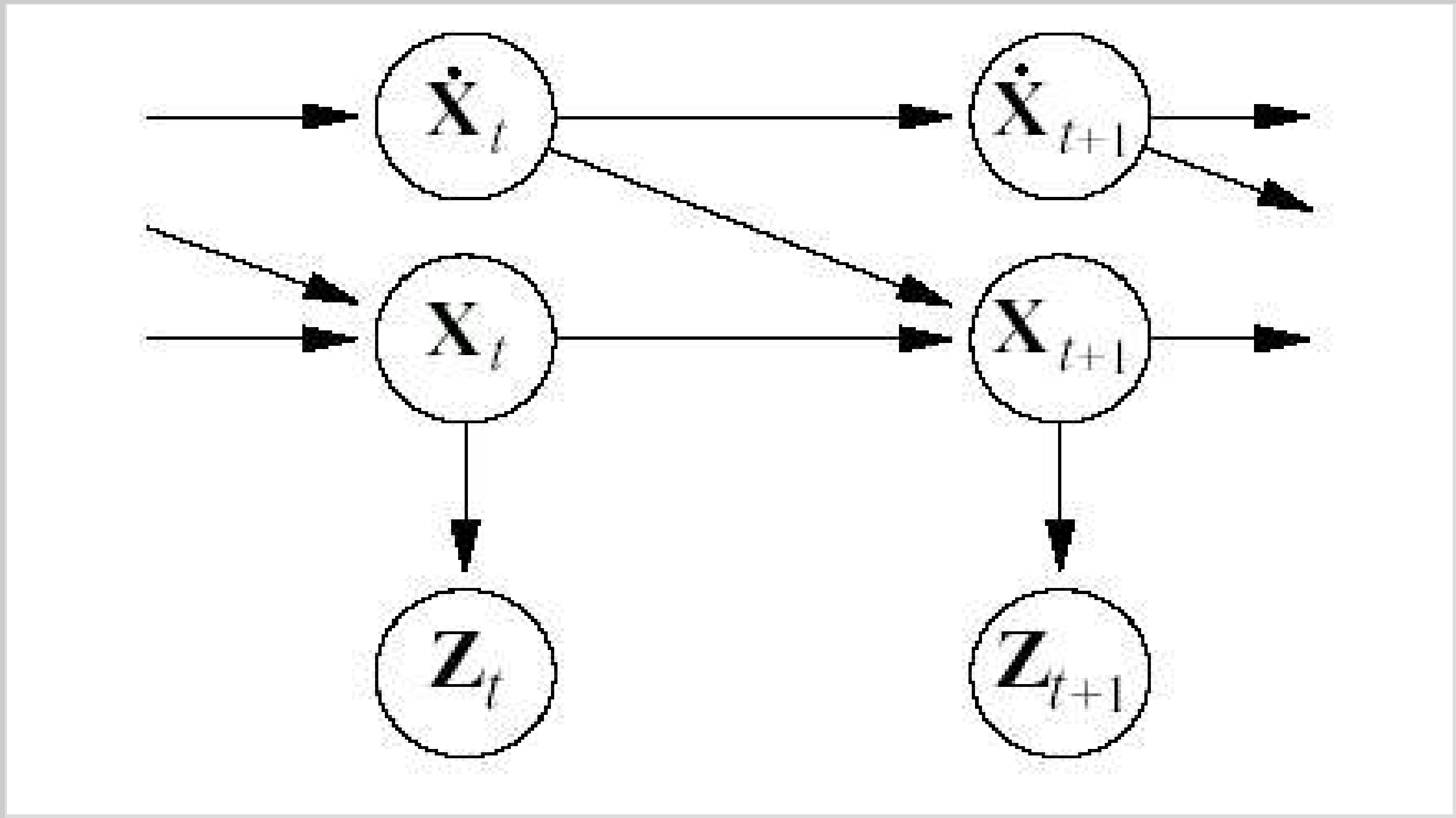
Hidden

Observed

Kalman Filter

- Modelling systems described by a set of continuous variables
  - Tracking a bird flying Xt = X,Y,Z, dX, dY, dZ
  - Airplanes, robots, ecosystems...
- Gaussian prior, linear Gaussian transition model and sensor model
  - i.e next state is a linear function of current state plus some Gaussian noise
  - Key property: linear Gaussian family remains closed under standard Baysian network operations

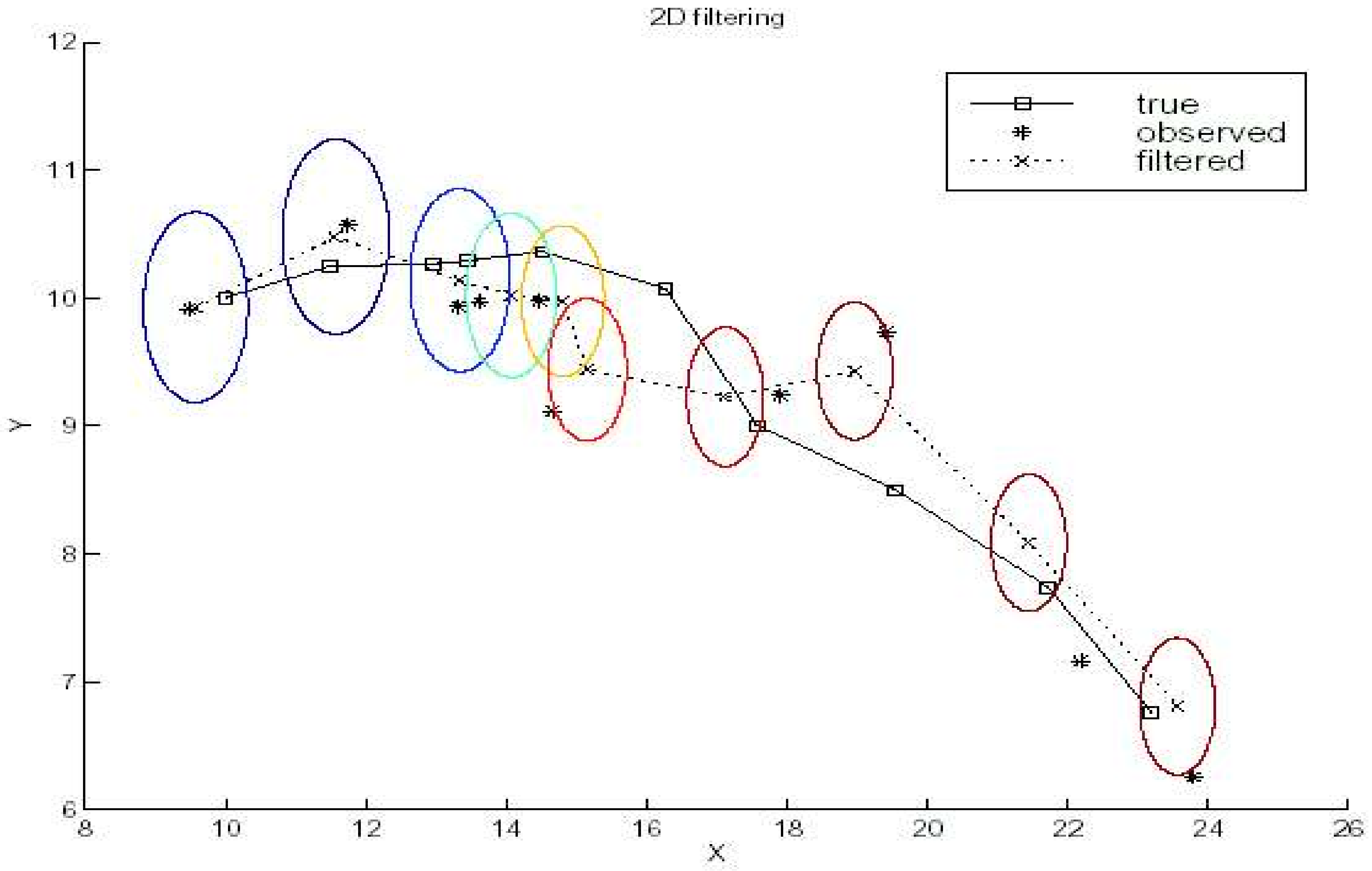
Kalman Filter



Position, velocity and position measurement

Basically we forward messages (means + covariance matrix) to produce new message (means + covariance matrix)

Kalman filtering



Kalman Smoothing