

430.457

Introduction to Intelligent Systems

Prof. Songhwai Oh
ECE, SNU

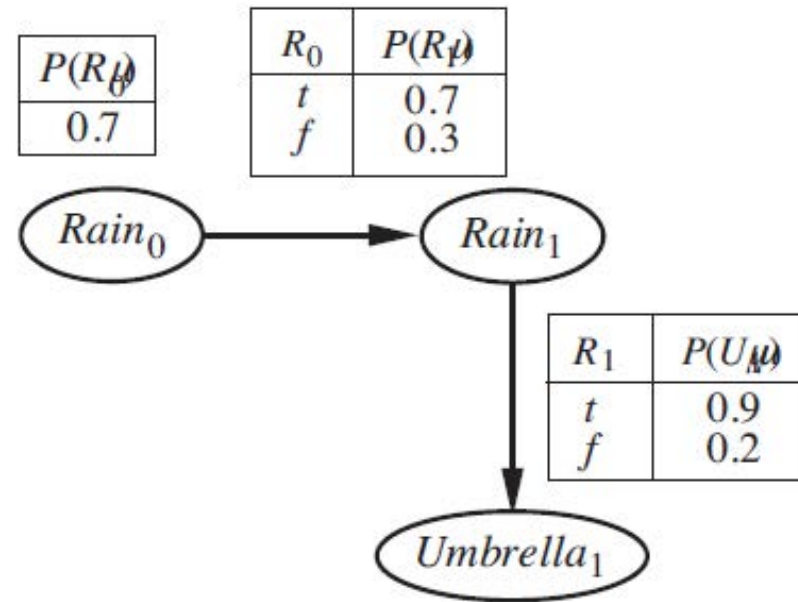
DYNAMIC BAYESIAN NETWORKS

Dynamic Bayesian Networks (DBNs)

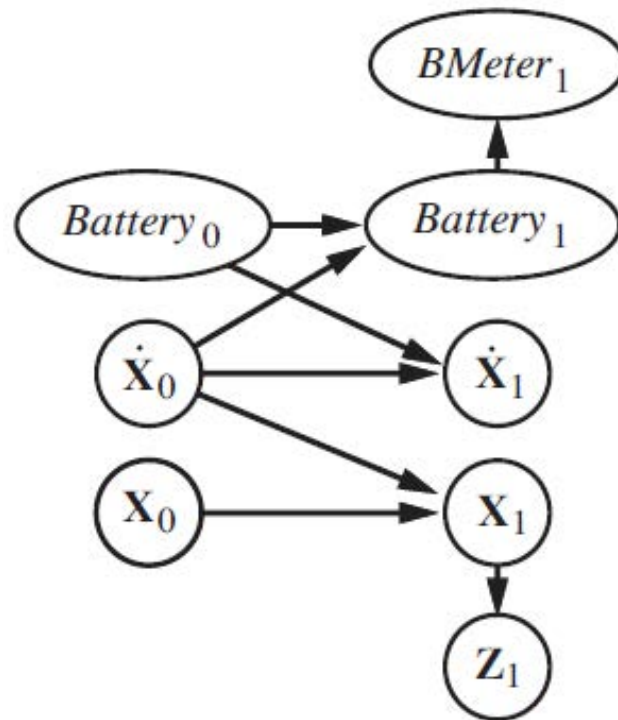
- Bayesian network for representing a temporal probability model
 - Includes hidden Markov Models and Kalman filters
- Decomposition of hidden variables
 - Example:
 - A DBN with 20 Boolean variables; each variable has three parents in the previous time; The transition model has $20 \times 2^3 = 160$ probabilities.
 - A HMM will require 2^{40} probabilities in its transition model (there are 2^{20} states).

Constructing DBNs

- Initial state distribution: $P(\mathbf{X}_0)$
- Transition model: $P(\mathbf{X}_{t+1}|\mathbf{X}_t)$
- Sensor model: $P(\mathbf{E}_t|\mathbf{X}_t)$
- Stationary process



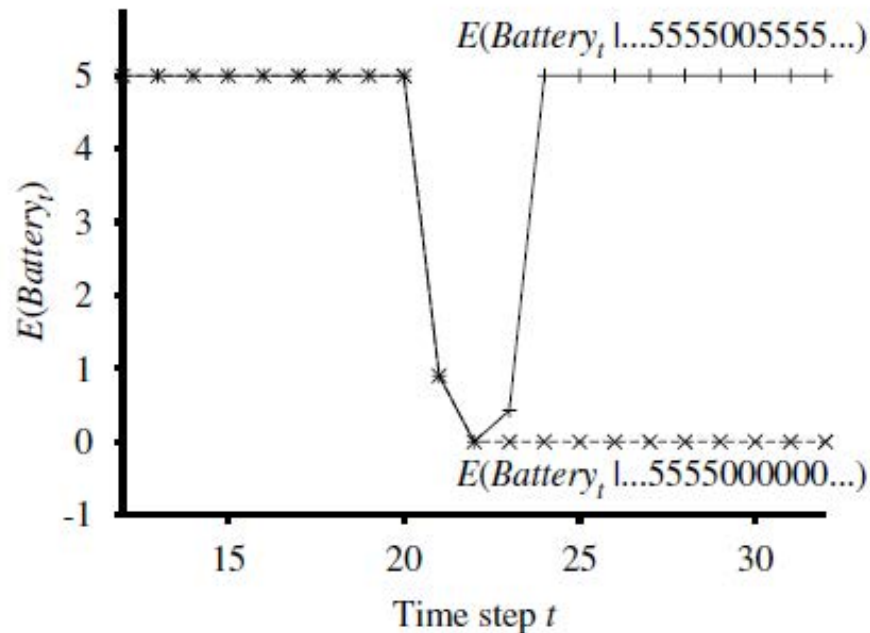
Example: Mobile Robot



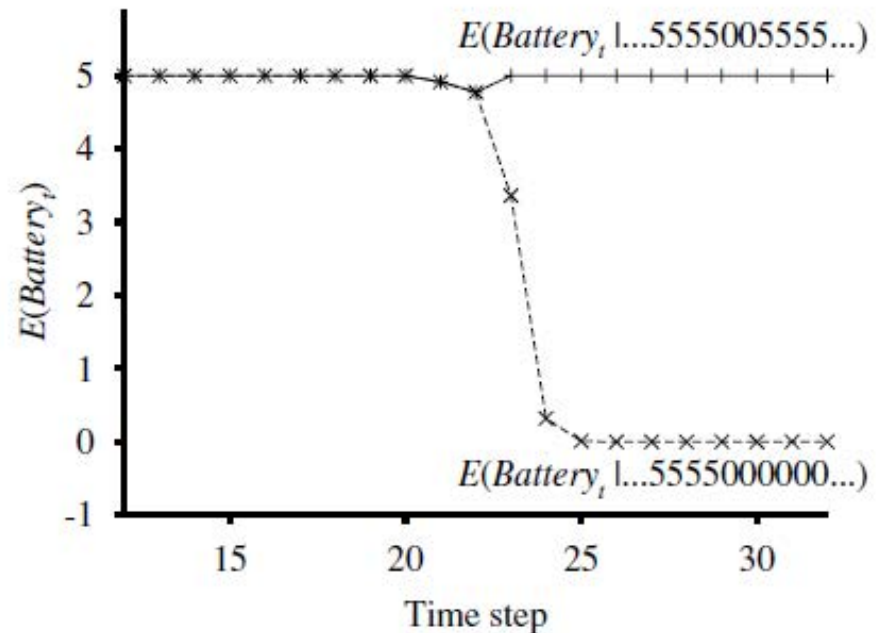
1. Sensor noise
2. Sensor failures
 - Transient failure
 - Persistent failure

Transient Failure Model

Transient failure model: $P(BMeter_t = 0 \mid Battery_t = 5) = 0.03$

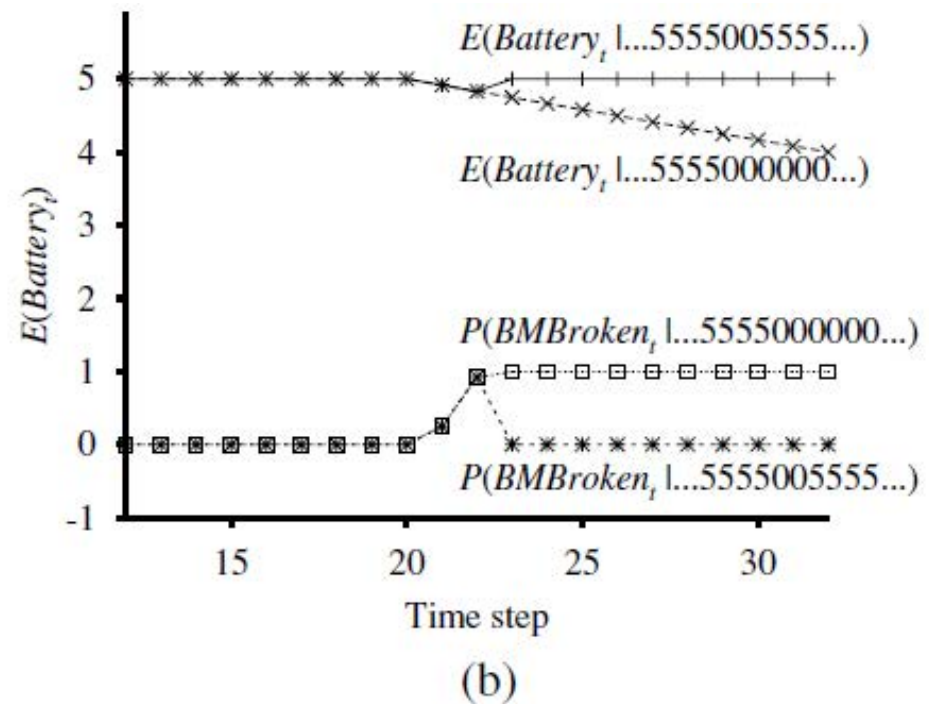
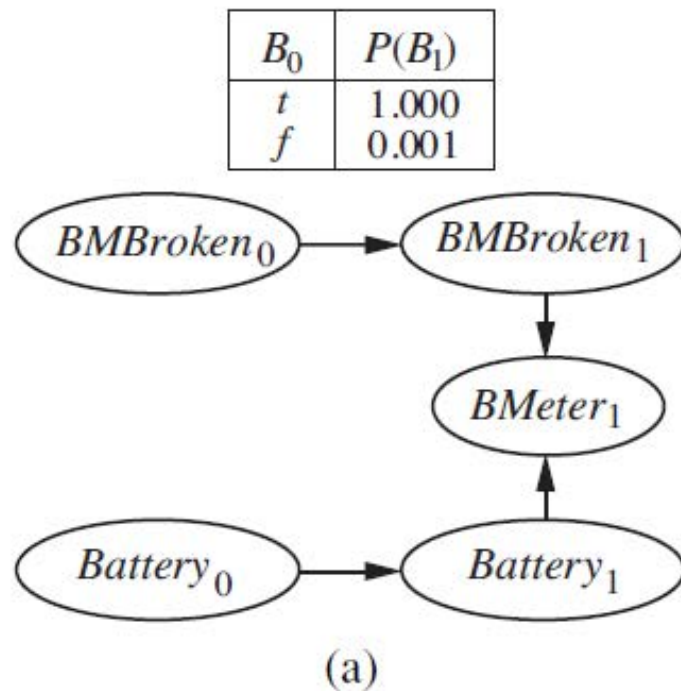


without the transient failure model



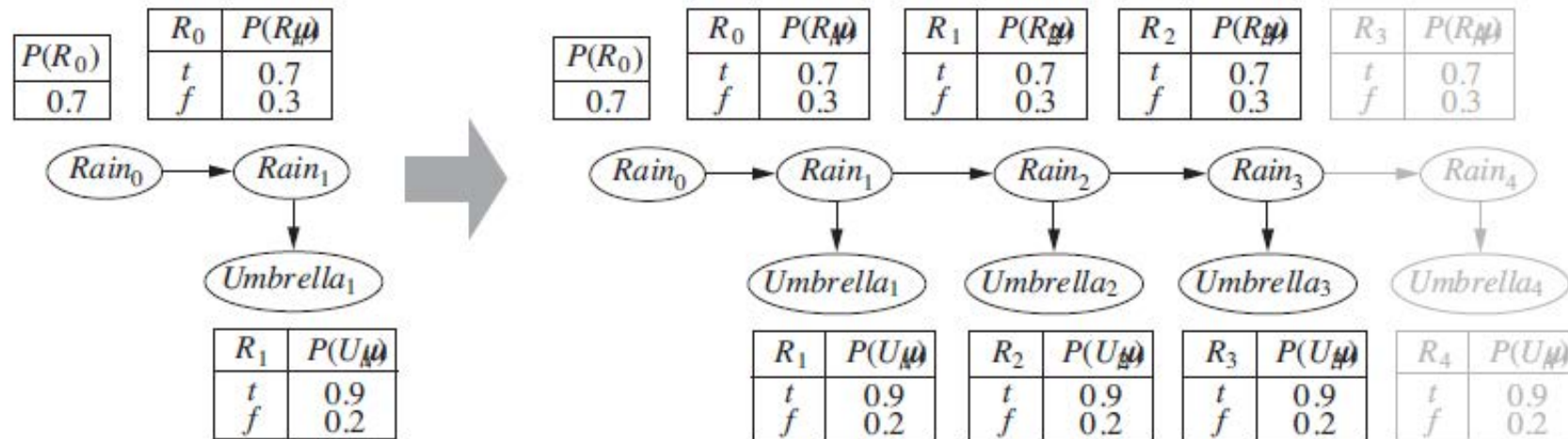
with the transient failure model

Persistent Failure Model



Exact Inference in DBNs

Unrolling



- We can apply any exact inference algorithm to the unrolled dynamic Bayesian network
 - E.g., variable elimination, clustering, ...
- But, in general, the complexity is exponential in the number of states.
- Infeasible for large number of variables

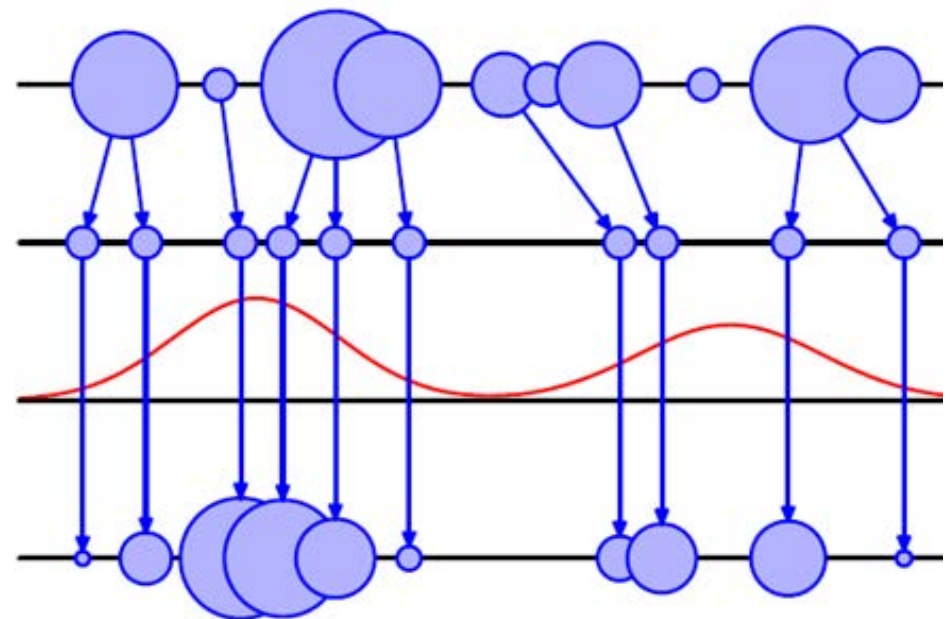
Particle Filtering: Approximate Inference in DBNs

Also known as sequential Monte Carlo (SMC):

- Sequential importance sampling (SIS)
- Sequential importance resampling (SIR)

Other names:

- bootstrap filtering
- condensation (computer vision)
- interacting particle approximation
- survival of the fittest



Particle Filtering

Sample N particles from $P(\mathbf{X}_0)$.

1. Each sample is propagated forward by sampling the next state value $\mathbf{x}_{t+1}$ given the current value $\mathbf{x}_t$ for the sample, based on the transition model $\mathbf{P}(\mathbf{X}_{t+1} | \mathbf{x}_t)$.
2. Each sample is weighted by the likelihood it assigns to the new evidence, $P(\mathbf{e}_{t+1} | \mathbf{x}_{t+1})$.
3. The population is *resampled* to generate a new population of N samples. Each new sample is selected from the current population; the probability that a particular sample is selected is proportional to its weight. The new samples are unweighted.

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

Example

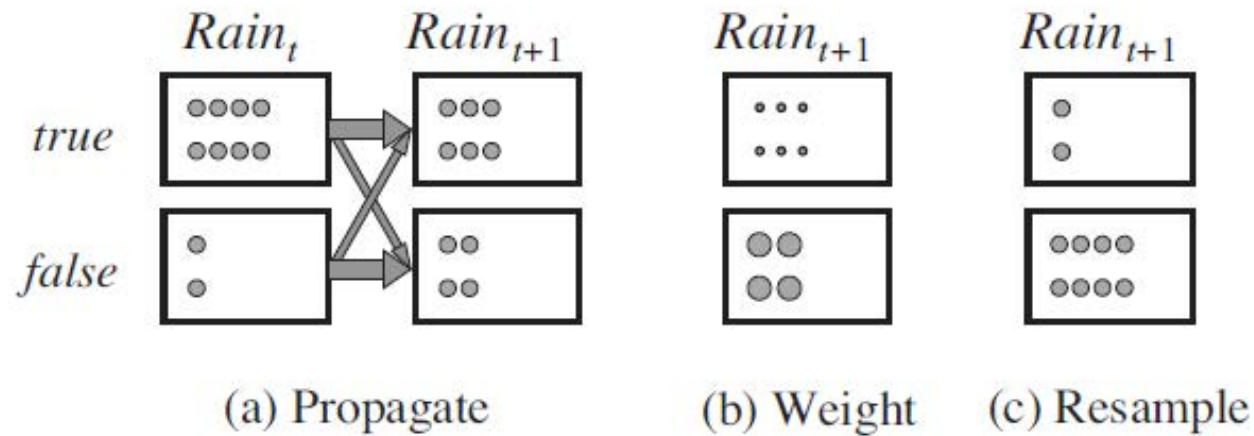


Figure 15.18 The particle filtering update cycle for the umbrella DBN with $N = 10$, showing the sample populations of each state. (a) At time t , 8 samples indicate *rain* and 2 indicate $\neg rain$. Each is propagated forward by sampling the next state through the transition model. At time $t + 1$, 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 $\neg rain$.

Consistency of Particle Filtering

- Let $N(\mathbf{x}_t|\mathbf{e}_{1:t})$ be the number of samples occupying state $\mathbf{x}_t$ given $\mathbf{e}_{1:t}$.
- Then, $\frac{1}{N}N(\mathbf{x}_t|\mathbf{e}_{1:t}) \approx P(\mathbf{x}_t|\mathbf{e}_{1:t})$ for large N .
- Sampling from the transition model:

$$N(\mathbf{x}_{t+1}|\mathbf{e}_{1:t}) = \sum_{\mathbf{x}_t} P(\mathbf{x}_{t+1}|\mathbf{x}_t)N(\mathbf{x}_t|\mathbf{e}_{1:t})$$

- Weighting by the likelihood:

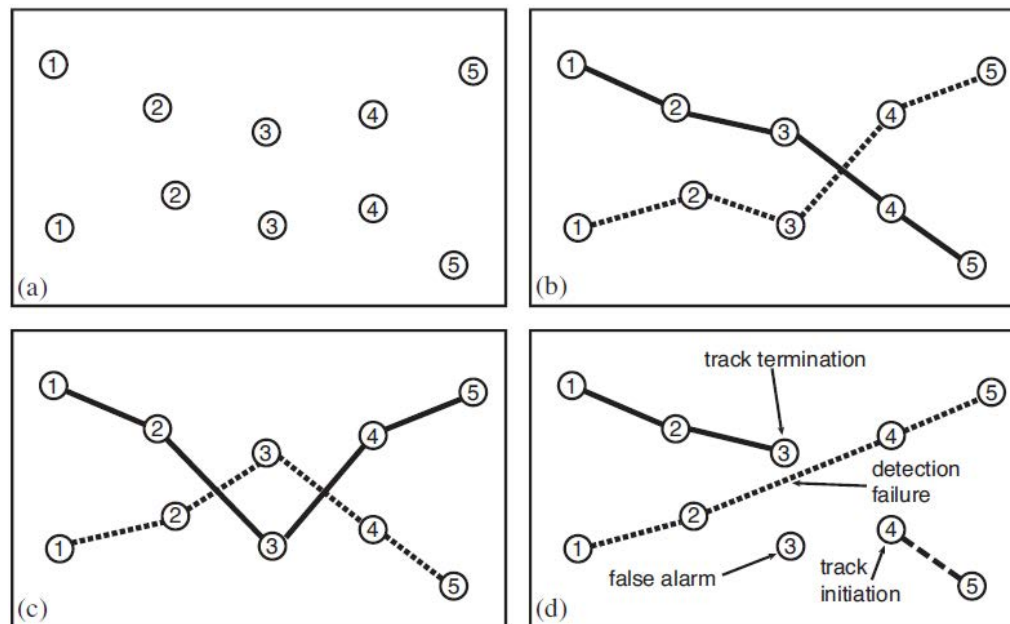
$$W(\mathbf{x}_{t+1}|\mathbf{e}_{1:t+1}) = P(\mathbf{e}_{t+1}|\mathbf{x}_{t+1})N(\mathbf{x}_{t+1}|\mathbf{e}_{1:t})$$

Hence,

$$\begin{aligned}\frac{1}{N}N(\mathbf{x}_{t+1}|\mathbf{e}_{1:t+1}) &= \alpha W(\mathbf{x}_{t+1}|\mathbf{e}_{1:t+1}) = \alpha P(\mathbf{e}_{t+1}|\mathbf{x}_{t+1})N(\mathbf{x}_{t+1}|\mathbf{e}_{1:t}) \\ &= \alpha P(\mathbf{e}_{t+1}|\mathbf{x}_{t+1}) \sum_{\mathbf{x}_t} P(\mathbf{x}_{t+1}|\mathbf{x}_t)N(\mathbf{x}_t|\mathbf{e}_{1:t}) \\ &\approx \alpha N P(\mathbf{e}_{t+1}|\mathbf{x}_{t+1}) \sum_{\mathbf{x}_t} P(\mathbf{x}_{t+1}|\mathbf{x}_t)P(\mathbf{x}_t|\mathbf{e}_{1:t}) \\ &= \alpha' P(\mathbf{e}_{t+1}|\mathbf{x}_{t+1}) \sum_{\mathbf{x}_t} P(\mathbf{x}_{t+1}|\mathbf{x}_t)P(\mathbf{x}_t|\mathbf{e}_{1:t}) \\ &= P(\mathbf{x}_{t+1}|\mathbf{e}_{1:t+1})\end{aligned}$$

Tracking Many Objects

- When there are many objects and many measurements, we may not know which object generated which observation.
- This problem is known as the **identity management** or **data association** problem.



$$P(x_{0:t}^A, x_{0:t}^B, e_{1:t}^1, e_{1:t}^2) = P(x_0^A)P(x_0^B) \prod_{i=1}^t P(x_i^A | x_{i-1}^A)P(x_i^B | x_{i-1}^B) P(e_i^1, e_i^2 | x_i^A, x_i^B)$$

$$P(x_{0:t}^A, x_{0:t}^B, e_{1:t}^1, e_{1:t}^2) = P(x_0^A)P(x_0^B) \prod_{i=1}^t P(x_i^A | x_{i-1}^A)P(x_i^B | x_{i-1}^B) P(e_i^1, e_i^2 | x_i^A, x_i^B)$$

$$\begin{aligned} P(e_i^1, e_i^2 | x_i^A, x_i^B) &= \sum_{\omega_i} P(e_i^1, e_i^2 | x_i^A, x_i^B, \omega_i) P(\omega_i | x_i^A, x_i^B) \\ &= \sum_{\omega_i} P(e_i^{\omega_i(A)} | x_i^A) P(e_i^{\omega_i(B)} | x_i^B) P(\omega_i | x_i^A, x_i^B) \\ &= \frac{1}{2} \sum_{\omega_i} P(e_i^{\omega_i(A)} | x_i^A) P(e_i^{\omega_i(B)} | x_i^B) . \end{aligned}$$

- ω_i is the assignment between objects and observations. For n objects, there are $n!$ possible values for ω_i .
- **Hungarian algorithm** finds the best assignment.
- For handling data association problems for multiple time steps, more sophisticated algorithms exist.