Raciocínio Probabilistico

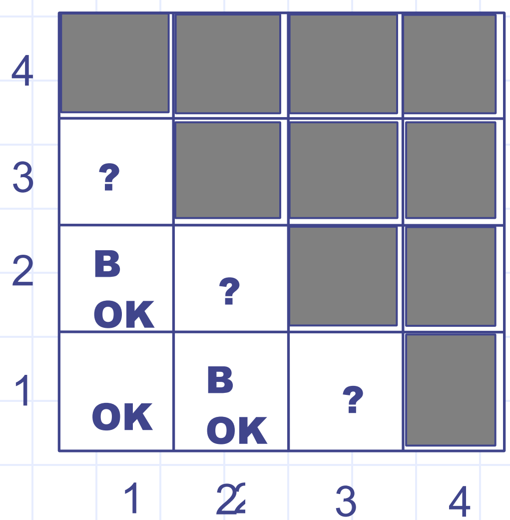
Métodos de Computação Inteligente 2006.1

Ricardo Scholz e Vitor Dantas
<reps, vcd>@cin.ufpe.br

Motivação

- Agentes precisam lidar com incertezas
  - Ambientes
    - Não determinísticos
    - Parcialmente observáveis

- Exemplo: Wumpus



$$
\begin{aligned}
&Breeze(2,1) \wedge Breeze(1,2) \Rightarrow \\
&((Pit(3,1) \wedge Pit(2,2) \wedge Pit(1,3)) \\
&\vee (Pit(3,1) \wedge Pit(2,2)) \\
&\vee (Pit(2,2) \wedge Pit(1,3)) \\
&\vee (Pit(3,1) \wedge Pit(1,3)) \\
&\vee Pit(2,2))
\end{aligned}
$$

Limitações da lógica para representar conhecimento incerto

- Engajamento epistemológico: crença sobre fato do mundo representado como fórmula lógica
  - certamente verdadeira
  - certamente falsa
  - totalmente desconhecida
- Incertezas representáveis apenas através da disjunção
  - É muito custoso modelar todos os casos possíveis.
  - Crenças iguais sobre todas as alternativas de uma disjunção.

Representando conhecimento incerto com Teoria da Probabilidade

- Variável Aleatória (Random Variable)
  - Um aspecto do mundo cujo "status" é desconhecido.
  - Variável assume valores dentro de um domínio.
  - Ex.:
    - domínio de Pit: <true, false>
    - domínio de Weather: <sunny, rainy, cloudy>
- Evento atômico
  - Especificação do valor de todas as variáveis

Representando conhecimento incerto com Teoria da Probabilidade

- Probabilidades
  - Expressam o grau de confiança do agente sobre aspectos do mundo.
    - 1 representa a certeza absoluta da verdade
    - 0 a certeza absoluta da falsidade
    - Valores entre 0 e 1 representam a confiança do agente
  - Ex.:
    - P(Weather = sunny) = 0.7
    - P(Pit = true) = 0.1
    - P(Pit = false) = 0.9
  - Igualmente:
    - P(sunny) = 0.7
    - P(pit) = 0.1
    - P(¬pit) = 0.9

Centro de Informática U·F·P·E

Distribuições de probabilidades

- Associam uma probabilidade a cada possível valoração de uma variável
  - Ex.:
    - domínio de Pit: <true, false>
    - **P**(Pit) = <0.1, 0.9>

    - domínio de Weather: <sunny, rainy, cloudy>
    - **P**(Weather) = <0.7, 0.1, 0.2>

Distribuição Conjunta de Probabilidades

- Define as probabilidades de todos os eventos atômicos possíveis.
  - Ex.: Domínio do dentista.

	dor de dente **toothache**		**¬toothache**	
	Boticão **catch**	**¬catch**	**catch**	**¬catch**
cárie **cavity**	0.108	0.012	0.072	0.008
¬cavity	0.016	0.064	0.144	0.576

Aprendizado de máquina - observações do ambiente

Probabilidades a priori e a posteriori

- A priori (incondicional)
  - Antes de se obter evidências.
- A posteriori (condicional)
  - Expressa o grau de confiança após alguma evidência ter sido obtida.
- Ex.:
  - $P(pit) = 0.1$
  - $P(pit \mid \neg breeze) = 0.05$

Probabilidade condicional e regra do produto

- Podemos definir uma probabilidade condicional em termos de probabilidades incondicionais.
  - $P(a \mid b) = P(a \wedge b) / P(b)$

- Regra do produto:
  - $P(a \wedge b) = P(a \mid b) P(b)$
  - $P(a \wedge b) = P(b \mid a) P(a)$

  Ou, genericamente:
  - $\mathbf{P}(X, Y) = \mathbf{P}(X \mid Y)\, \mathbf{P}(Y)$
    - Onde $P(X=x_i \mid Y=y_j)$, $\forall i,j$

Regra de Bayes

- $P(a \wedge b) = P(a \mid b)\, P(b)$
- $P(a \wedge b) = P(b \mid a)\, P(a)$

Finalmente, a regra de Bayes

- $P(b \mid a) = \boxed{P(a \mid b)\, P(b) \,/\, P(a)}$

Que generaliza para

- $\mathbf{P}(Y \mid X) = \mathbf{P}(X \mid Y)\, \mathbf{P}(Y) \,/\, \mathbf{P}(X)$

Ou ainda, quando há evidência

- $\mathbf{P}(Y \mid X, \mathbf{e}) = \mathbf{P}(X \mid Y, \mathbf{e})\, \mathbf{P}(Y \mid \mathbf{e}) \,/\, \mathbf{P}(X \mid \mathbf{e})$

Marginalização

	toothache		¬toothache	
	catch	¬catch	catch	¬catch
cavity	0.108	0.012	0.072	0.008
¬cavity	0.016	0.064	0.144	0.576

- $P(Y) = \Sigma_z P(Y, z)$
  - P(Cavity) = <0.108 + 0.012 + 0.072 + 0.008, 0.016 + 0.064 + 0.144 + 0.576> = <0.2, 0.8>

P(carieldorDeDente) = (0.108 + 0.012)/(0.108 + 0.012 + 0.016 + 0.064)

- Como calcular a probabilidade de uma proposição dada alguma evidência? Ex: P(Cavity|toothache)
  - Pela regra do produto:
    - P(cavity | toothache) = P(cavity ∧ toothache) / P(toothache)
    - P(¬cavity | toothache)=P(¬cavity ∧ toothache) / P(toothache)
    - 1/P(toothache) é apenas uma constante de normalização (α), logo basta calcular P(Cavity, toothache) utilizando marginalização

Problemas da inferência por marginalização

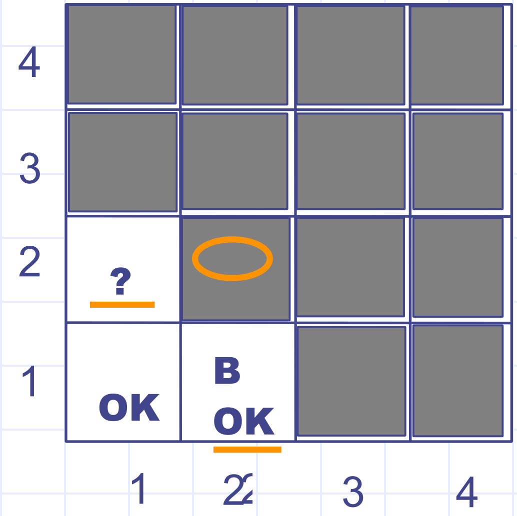
- Complexidade exponencial
  - No espaço
    - Para n variáveis (booleanas), a distribuição de probabilidade conjunta possui 2^n entradas
  - No tempo
    - No pior caso, a marginalização precisa somar todos os elementos da tabela de distribuição conjunta

Independência entre variáveis

- Independência absoluta
  - Ocorre quando uma variável não influencia outra de modo algum.
  - Ex.: **P**(Pit | Weather = rainy ∨ Weather = cloudy) = **P**(Pit)
- Independência condicional
  - Ocorre quando uma variável influencia outra, mas não há relação causal entre elas.
  - Dada a evidência sobre a real causa, não é mais necessário modelar tais influências indiretas.
  - Exemplo a seguir.

Independência condicional

- Exemplo:
  - ~~**P**(Breeze(1,2) | Breeze(2,1))~~
  - **P**(Breeze(1,2) | Breeze(2,1) ∧ Pit(2,2))
  - **P**(Breeze(1,2) | $\boxed{\text{Breeze(2,1)}}$ ∧ Pit(2,2)) = **P**(Breeze(1,2) | Pit(2,2))

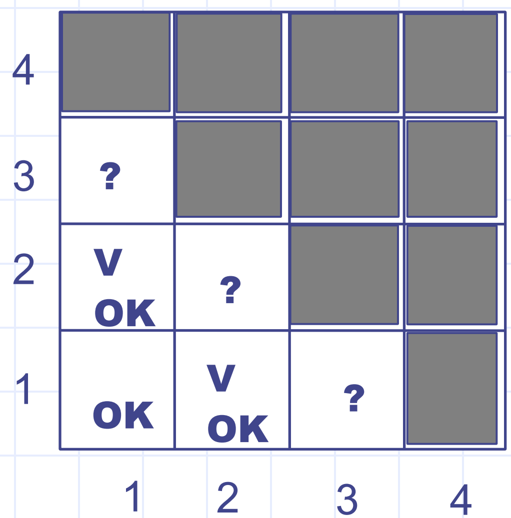


Redução da complexidade da distribuição conjunta

- Distribuição conjunta:
  - P(Toothache, Catch, Cavity) =
    (regra do produto)
  - P(Toothache, Catch | Cavity) P(Cavity) =
    (independência condicional)
  - P(Toothache | Cavity) P(Catch | Cavity) P(Cavity)
    - De 2^3-1 valores necessários para 2+2+1
    - Reduz de $O(2^n)$ para $O(n)$
  - De uma maneira geral (assumindo independência entre os efeitos, dada a causa)
    - $P(Causa, Efeito_1, Efeito_2,...)=P(Causa)\prod_i P(Efeito_i|Causa)$

Exemplo: Wumpus...

- Sejam:
  - P_{ij} variável booleana que indica se a posição (i,j) possui buraco
  - B_{ij} variável booleana que indica se a posição (i,j) possui brisa (usaremos só B_{11}, B_{12} e B_{21})



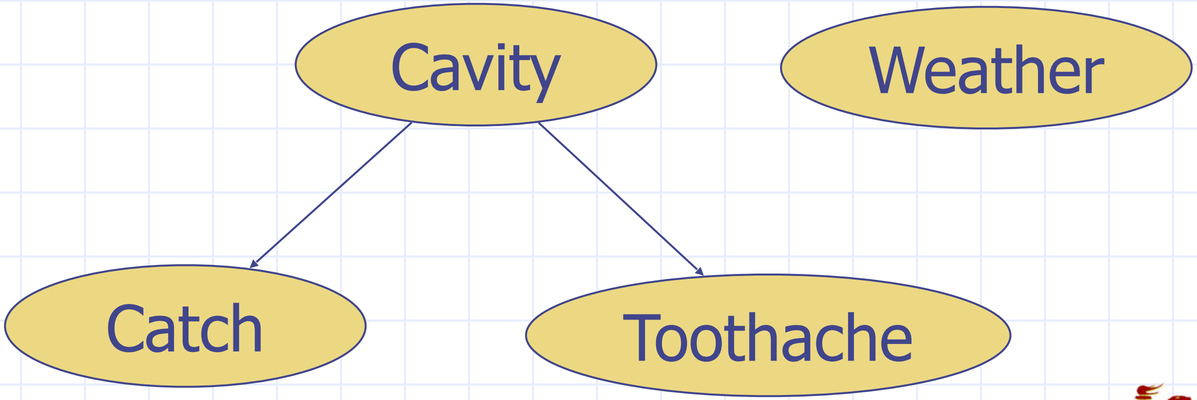
- Tell($\neg b_{11}$)
  Tell(b_{12})
  Tell(b_{21})
- Ask(P(P_{13} | KB))
  Ask(P(P_{22} | KB))
  Ask(P(P_{31} | KB))

Wumpus: Especificação da distribuição conjunta e inferência do melhor caminho

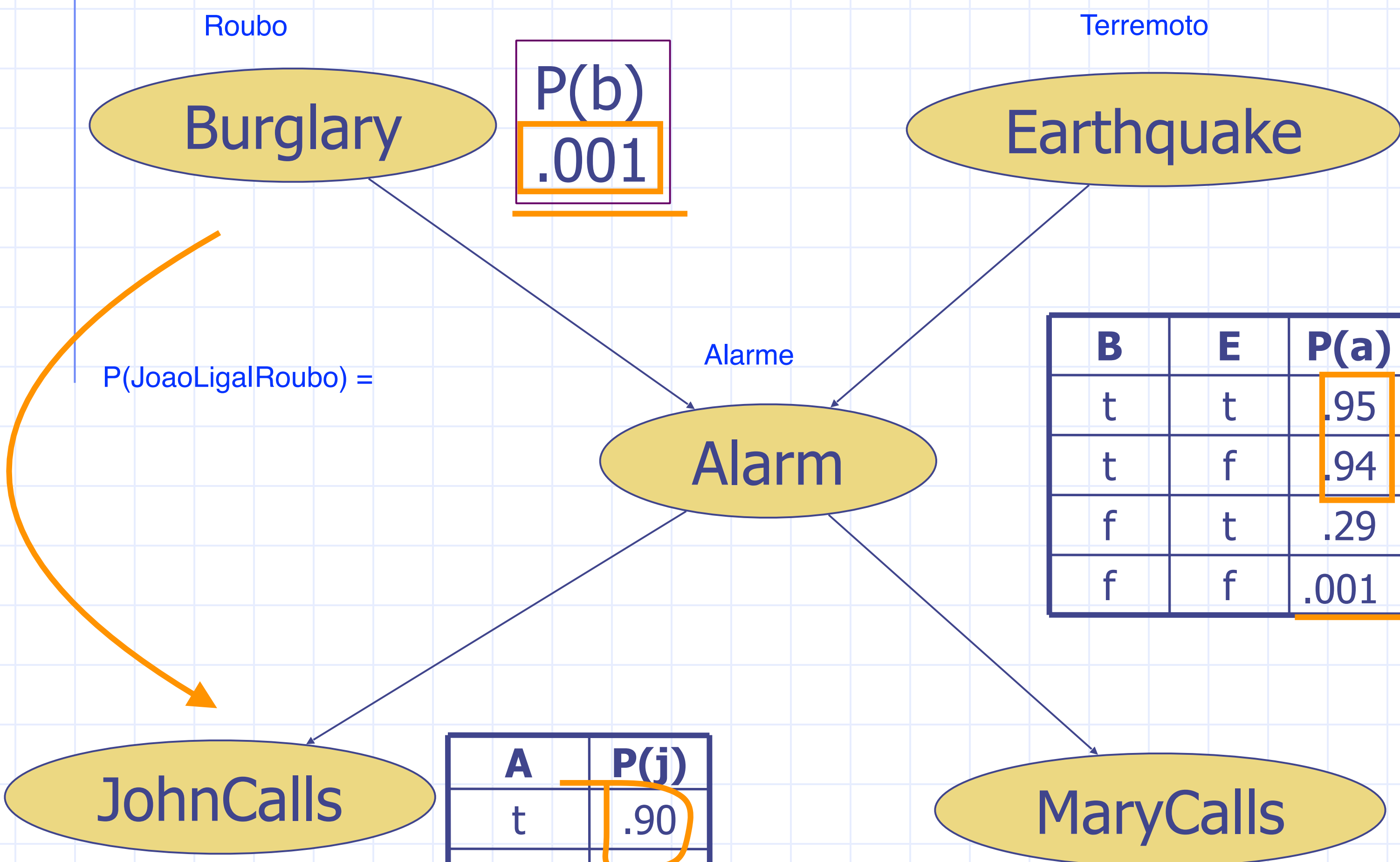
- **A DPC é dada por P**$(P_{11}, P_{12}, \ldots, P_{44}, B_{11}, B_{12}, B_{21})$
  - $\mathbf{P}(P_{11}, P_{12}, \ldots, P_{44}, B_{11}, B_{12}, B_{21})$
    $= \mathbf{P}(B_{11}, B_{12}, B_{21} \mid P_{11}, P_{12}, \ldots, P_{44})\ \mathbf{P}(P_{11}, P_{12}, \ldots, P_{44})$
- **Como $P(P_{ij})$ = <0.2, 0.8>, e as probabilidades de $P(P_{ij})$ são independetes**
  - $\mathbf{P}(P_{11}, P_{12}, \ldots, P_{44}) = 0.2^{nPits} \times 0.8^{16-nPits}$
- **Sabe-se um algoritmo para calcular** $\mathbf{P}(B_{11}, B_{12}, B_{21} | P_{11}, P_{12}, \ldots, P_{44})$
  - **Para toda variável P_{ij}, se P_{ij} = true**
    - **Para toda variável B, se B é adjacente a P_{ij} e B = false, retorne 0**
  - **Retorne 1**
- **Com a DPC, a máquina de inferência pode responder a** Ask($P(P_{13} \mid KB)$) **por marginalização**
  - $P(P_{13} \mid KB) = P(P_{31} \mid KB) = 31\%$
  - $P(P_{22} \mid KB) = 86\%$
    - O agente deve seguir por (1,3) ou (3,1)

Redes Bayesianas para representação de conhecimento incerto

- Rede bayesiana (sintaxe)
  - Variáveis são representadas por nós
  - O grafo formado não possui ciclos
  - Se há uma seta de A para B, então A (pai) possui uma influência direta sobre B



Redes bayesianas e tabela de probabilidades condicionais



P(b)
.001

P(e)
.002

B	E	P(a)
t	t	.95
t	f	.94
f	t	.29
f	f	.001

A	P(j)
t	.90
f	.05

A	P(m)
t	.70
f	.01

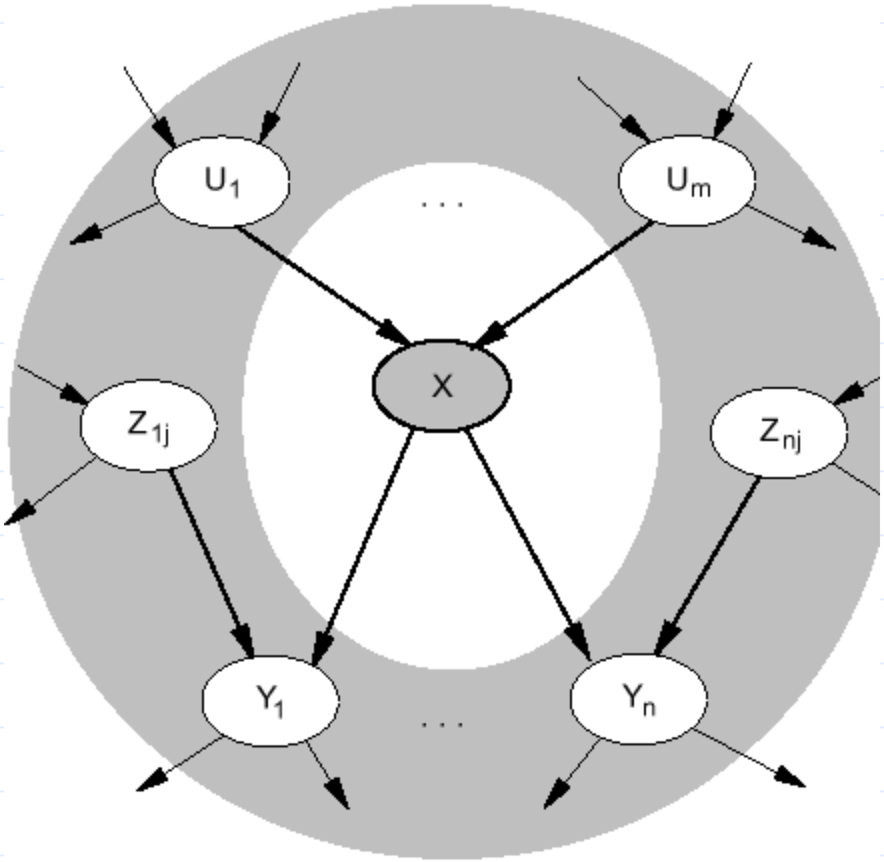
Semântica das Redes bayesianas

- Representa a distribuição conjunta
  - $P(x_1, \ldots, x_n) = \Pi_{i=1|n} P(x_i|parents(X_i))$
  - Ex.
    - $P(j \wedge m \wedge a \wedge \neg b \wedge \neg e) =$
      $P(j|a)\ P(m|a)\ P(a|\neg b \wedge \neg e)\ P(\neg b)\ P(\neg e) =$
      0.00062
  - Complexidade $O(n2^k)$ contra $O(2^n)$ da tabela de distribuição conjunta, onde k é o número máximo de pais por nó
    - No exemplo do alarme, representa 10 valores contra 2^5-1 (31) da representação pela tabela de DPC

P(jlb) = P(jla) P(alb ^ ~e) P(b)

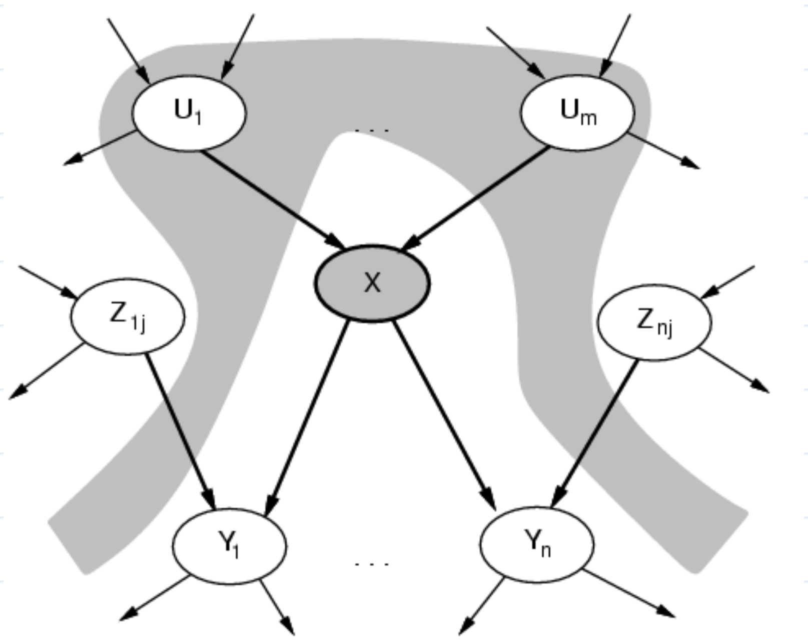
Propriedades das redes bayesianas

- Um nó é condicionalmente independente de todos os outros nós da rede, dado seus pais, filhos e pais dos filhos (markov blanket)



Propriedades das redes bayesianas

- Um nó é condicionalmente independente dos não-descendentes, dado seus pais
  - Ex. P(MaryCalls | JohnCalls, Alarm, Earthquake, Burglary) = P(MaryCalls | Alarm)



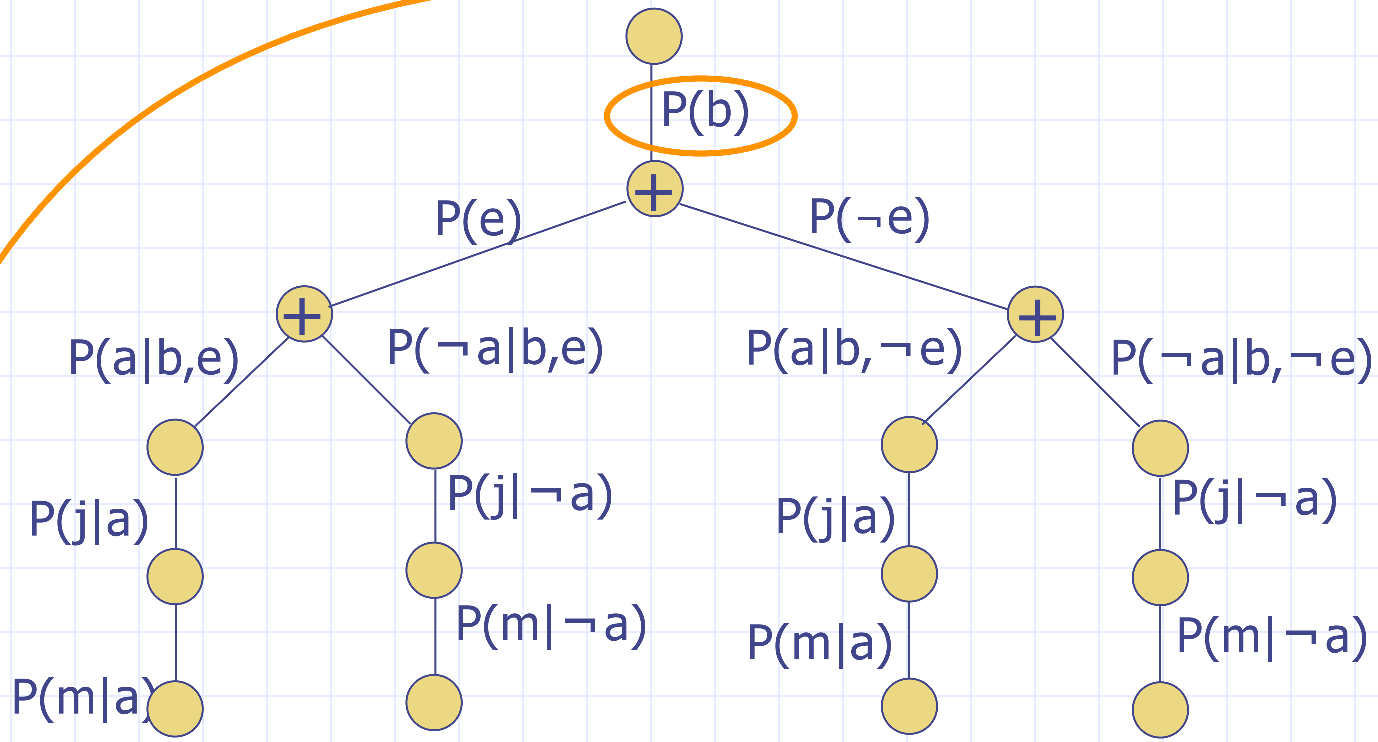
Redes Bayesianas para variáveis contínuas

- Discretizar os valores das variáveis aleatórias
- Especificar as funções de distribuição de probabilidade, com número pequeno de parâmetros (função linear gaussiana é a mais utilizada)

Inferência exata em redes bayesianas

- Como uma rede bayesiana representa a distribuição conjunta, o mesmo processo pode ser utilizado
- $P(B|j,m) = \alpha P(B,j,m) = \alpha \Sigma_t \Sigma_a P(B,e,a,j,m)$
  - Para B = true:
    - $P(b|j,m) = \alpha\ \Sigma_t\ \Sigma_a\ P(b)P(e)P(a|b,e)P(j|a)P(m|a)$
  - Porém, mais eficiente:
    - $P(b|j,m) = \alpha\ P(b)\Sigma_t P(e)\Sigma_a P(a|b,e)P(j|a)P(m|a)$

Árvore de expressões



$$P(b|j,m) = \alpha\, P(b)\Sigma_t P(e)\Sigma_a P(a|b,e)P(j|a)P(m|a)$$

Eliminação de variáveis

- Algoritmo para inferência exata
  - Processa a árvore de expressões bottom-up, armazenando os valores calculados
  - Observação:
    - Todas as variáveis que não são ancestrais da variável de consulta ou das evidências podem ser removidas da inferência
    - $P(J|l) = \alpha P(l)\Sigma_t P(t)\Sigma_a P(a|l,t)P(J|a)\Sigma_m P(m|a)$
      - Mas $\Sigma_m P(m|a)$ é 1 por definição, logo pode ser eliminado

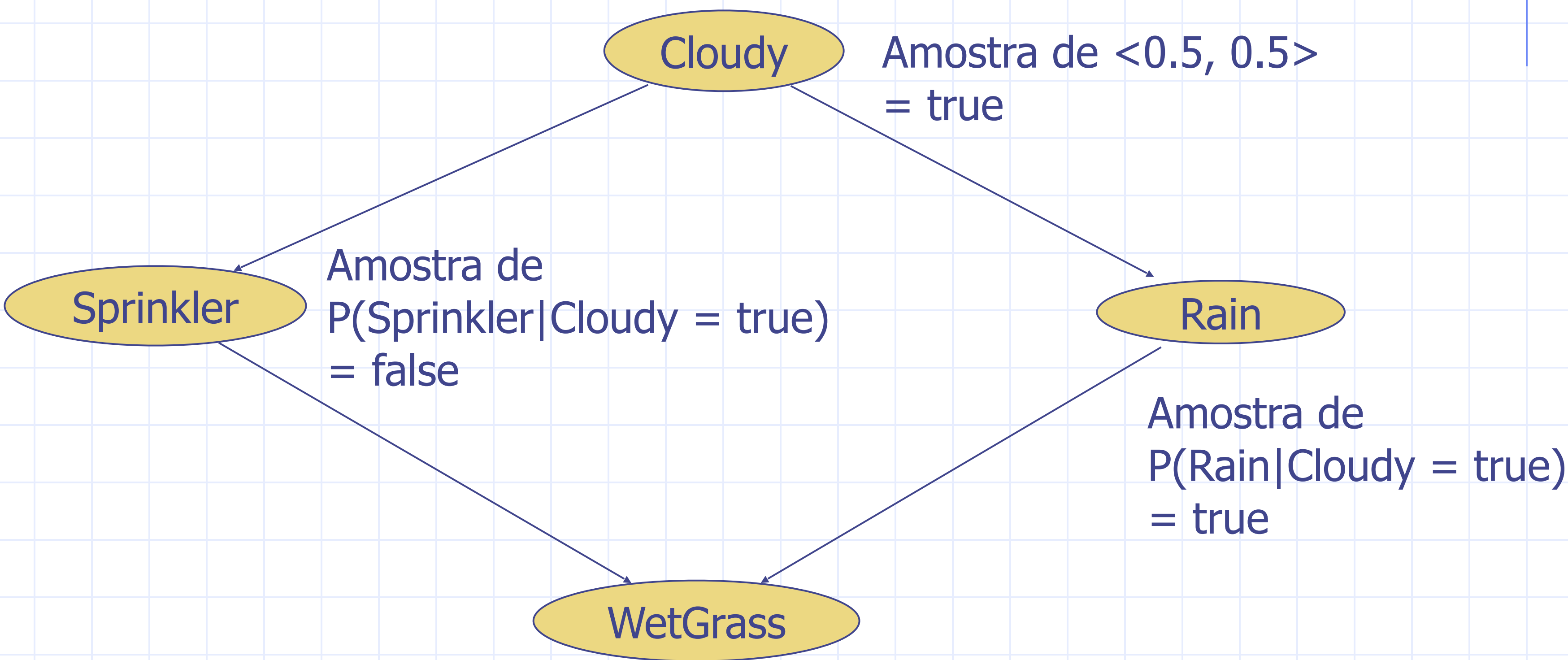
Complexidade do algoritmo de eliminação de variáveis

- Se a rede é uma polytree (no máximo um caminho entre dois nós)
  - linear no número de nós da rede
- Em geral, tempo e espaço exponencial (#P-Hard)

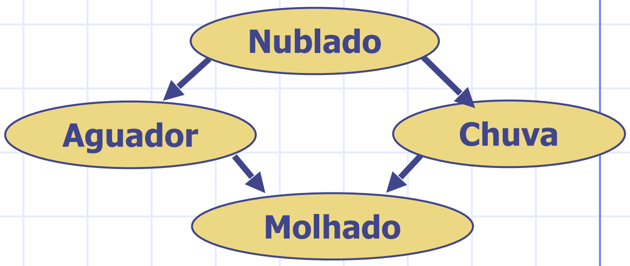
Inferência aproximada em redes bayesianas

- Inferência exata é intratável para redes grandes e muito conectadas
- Inferência aproximada permite obter uma solução mais eficiente, porém aproximada
  - A complexidade em tempo é dada pela qualidade da solução
- Escapa da NP-Completude, mas qualidade da solução diminui
- Baseada nos algoritmos de Monte Carlo

Amostragem direta



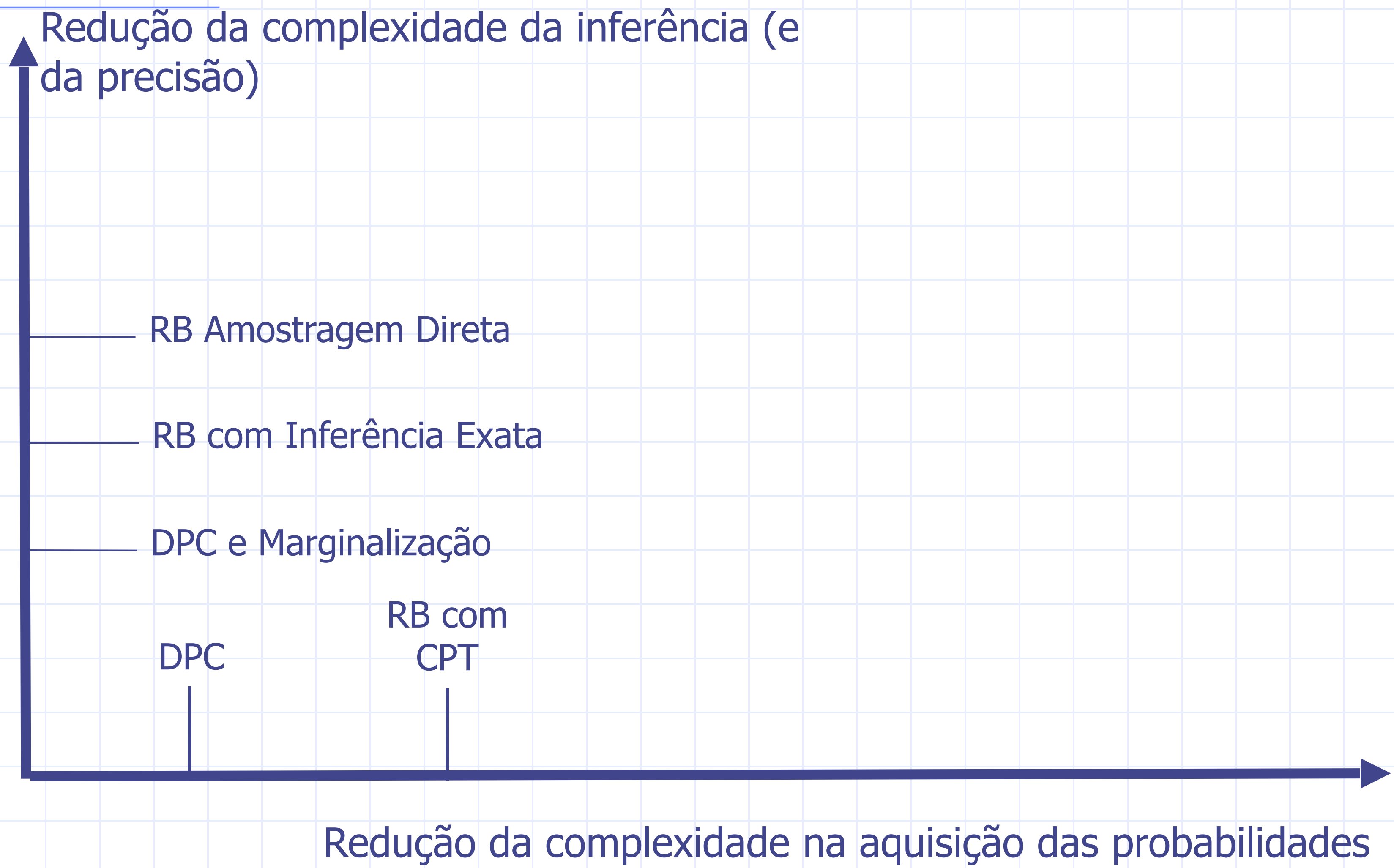
Evento atômico:
[true, false, true, true]



Amostragem direta

- $\lim_{N\to\infty}[N_{amostras}(x_1,\ldots,x_n)/N] = P(x_1,\ldots,x_n)$
- Quanto mais amostras, mais consistente a aproximação da DPC
- Exemplo:
  - Se em 1000 amostras, 511 delas tem Rain = true, P(rain) = 0.511

Resumo dos métodos de inferência dos agentes probabilistas



Raciocínio Probabilistico Temporal

Redes Bayesianas Dinâmicas

Raciocínio Probabilistico Temporal
Motivação

- Agentes em ambientes incertos têm que manter-se atualizados sobre o estado do ambiente;

- A tarefa fica mais difícil se considerarmos:
  - Percepções parciais e/ou ruidosas;
  - Incerteza sobre como o ambiente muda ao longo do tempo.

Raciocínio Probabilistico Temporal
Estados e Observações

- O processo de mudança do ambiente pode ser visto como uma série de "fatias de tempo";
- Cada instante contém um conjunto de variáveis randômicas, algumas observáveis, outras não;

Raciocínio Probabilistico Temporal
Notação

- X_t → Variável de estado S no tempo t
  - Ex.: $Chuva_1$, $Energia_5$
- E_t → Variável de evidência (observação) E no tempo t
  - Ex.: $GramaMolhada_1$, $LâmpadaAcesa_9$
- $X_{a:b}$ → Conjunto de variáveis de estado ou evidência de X_a até X_b
  - Ex.: $Chuva_{1:4}$ = $Chuva_1$, $Chuva_2$, $Chuva_3$, $Chuva_4$

Raciocínio Probabilistico Temporal

Algumas Definições Importantes

- Processo Estacionário
  - Processo de mudança governado por leis que não mudam ao longo do tempo
- Restrição sobre Observações
  - $P(E_t|X_{0:t}, E_{0:t-1}) = P(E_t|X_t)$

Raciocínio Probabilistico Temporal
Algumas Definições Importantes

- Premissa de Markov (Markov Assumption)
  - Estado atual depende de um histórico finito de estados anteriores:
    - $\forall t, P(X_t|X_{0:t-1}) = P(X_t|X_{t-n:t-1})$, sendo $n \geq 1$
  - Processos de Markov ou Cadeias de Markov
  - Processo de Markov de 1ª ordem (n = 1)
    - $\forall t, P(X_t|X_{0:t-1}) = P(X_t|X_{t-1})$

Raciocínio Probabilistico Temporal

Representação com Redes Bayesianas

Prior Distribution

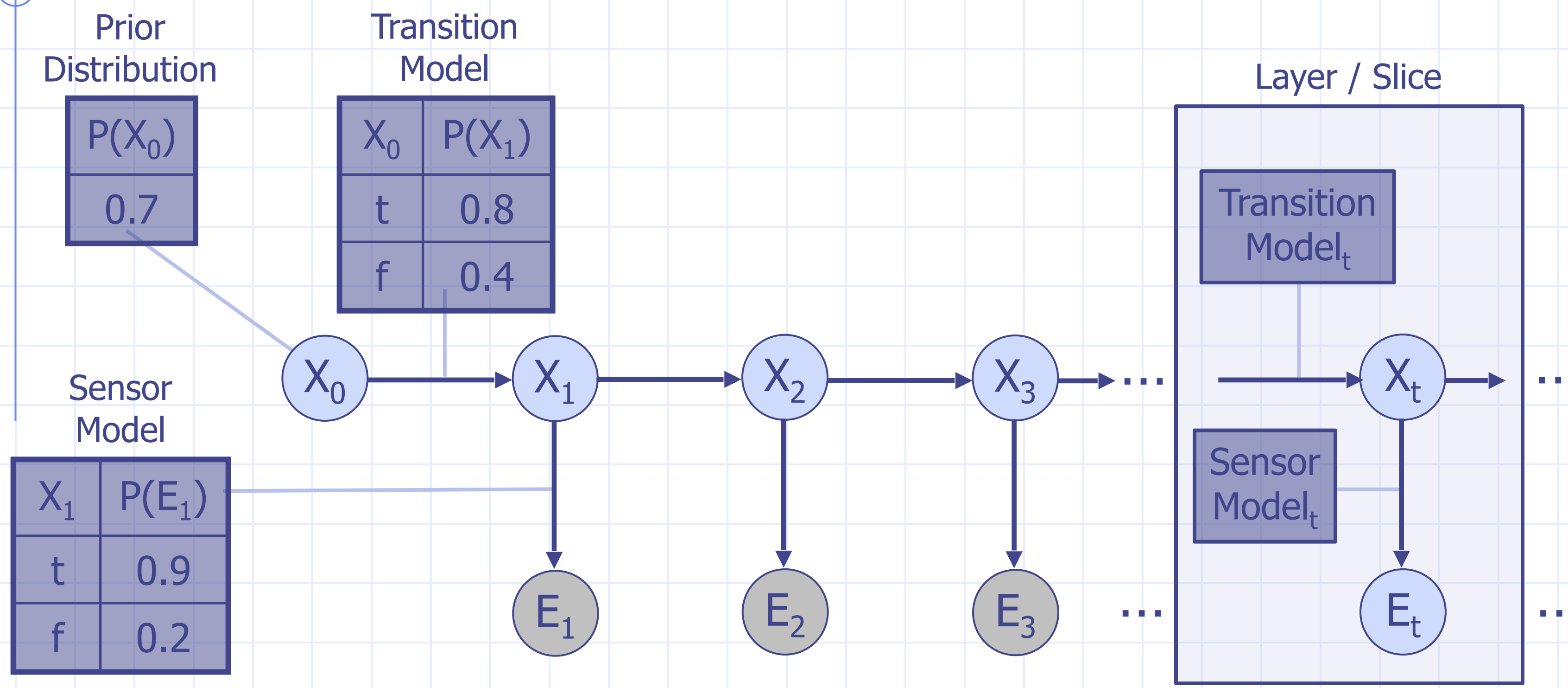
$P(X_0)$
0.7

Transition Model

X_0	$P(X_1)$
t	0.8
f	0.4

Sensor Model

X_1	$P(E_1)$
t	0.9
f	0.2



Complete Joint Distribution (Processo de Markov de 1ª ordem)

$P(X_0, X_1, \ldots, X_t, E_1, \ldots, E_t) = P(X_0)\prod P(X_i|X_{i-1})P(E_i|X_i)$, para i de 1 a t

Inferência em Modelos Temporais
Serviços de Raciocínio Probabilístico Temporal

- Filtering / Monitoring
  - $P(X_t|e_{1:t})$
- Prediction
  - $P(X_{t+k}|e_{1:t})$, para algum $k > 0$
- Smoothing / Hindsight
  - $P(X_k|e_{1:t})$, para $0 \leq k < t$
- Explicação Mais Provável
  - $\text{argmax}_{X_{1:t}} P(X_{1:t}|e_{1:t})$

Inferência em Modelos Temporais
Filtering :: Métodos para Inferência Temporal

- Estimativa por Recursão

$P(X_{t+1}|e_{1:t+1}) = f(e_{t+1}, P(X_t|e_{1:t})) = f(e_{t+1}, f(e_t, P(X_{t-1}|e_{1:t-1})))...$

$P(X_{t+1}|e_{1:t+1}) = P(X_{t+1}|e_{1:t}, e_{t+1})$

$= \alpha P(e_{t+1}|X_{t+1}, e_{1:t})P(X_{t+1}|e_{1:t})$ [by Bayes Rule]

$= \alpha P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$ [by previous assumtions]

$= \alpha P(e_{t+1}|X_{t+1})\Sigma_{xt}P(X_{t+1}|x_t, e_{1:t})P(x_t|e_{1:t})$

$= \alpha \mathbf{P(e_{t+1}|X_{t+1})\Sigma_{x_t}P(X_{t+1}|x_t)P(x_t|e_{1:t})}$

Inferência em Modelos Temporais

Filtering :: Métodos para Inferência Temporal

$$P(X_{t+1}|e_{1:t+1}) = \alpha \underbrace{\boxed{P(e_{t+1}|X_{t+1})}}_{\text{Sensor Model}} \Sigma_{xt} \underbrace{\boxed{P(X_{t+1}|x_t)}}_{\text{Transition Model}} P(x_t|e_{1:t})$$

$P(X_t|e_{1:t})$ pode ser visto como uma "mensagem" $f_{1:t}$ que é propagada atravez da seqüência, modificada a cada transição e atualizada a cada nova observação

$$P(X_t|e_{1:t}) = f_{1:t+1} = \alpha\text{FORWARD}(f_{1:t}, e_{t+1})$$

- Quando as variáveis de estado são discretas, o tempo e memória gastos em cada atualização são constantes!!!

Inferência em Modelos Temporais
Prediction :: Métodos para Inferência Temporal

- Predição pode ser encarado como Filtering sem a adição de uma nova observação:

$$P(X_{t+k+1} | e_{1:t}) = \Sigma_{x_{t+k}} P(X_{t+k+1} | x_{t+k}) P(x_{t+k} | e_{1:t})$$

- O que acontece se tentarmos prever cada vez mais longe?
  - A predição converge para um ponto fixo

Inferência em Modelos Temporais

Smoothing :: Métodos para Inferência Temporal

- Análogo a Filtering

$$
\begin{aligned}
P(X_k|e_{1:t}) &= P(X_k|e_{1:k}, e_{k+1:t}) \\
&= \alpha P(X_k|e_{1:k})P(e_{k+1:t}|X_k, e_{1:k}) \\
&= \alpha P(X_k|e_{1:k})P(e_{k+1:t}|X_k) \\
&= \alpha f_{1:k} b_{k+1:t}
\end{aligned}
$$

- A função BACKWARD é definida também analogamente:

$$b_{k+1:t} = \text{BACKWARD}(b_{k+2:t}, e_{k+1:t})$$

Inferência em Modelos Temporais
Tempo e Complexidade

- Tanto o FORWARD quanto o BACKWARD tem custo constante para um realizar um passo;
- Sendo assim, a complexidade de realizar Smoothing para uma fatia de tempo k com respeito à observação $e_{1:t}$ é O(t);
- Logo, para toda a seqüência, temos $0(t^2)$;
  - Para conseguir O(t) usa-se programação dinânica, evitando o cálculo repetido da mensagem propagada;
  - O algoritmo é chamado de FORWARD-BACKWARD

Inferência em Modelos Temporais
Seqüência Mais Provável

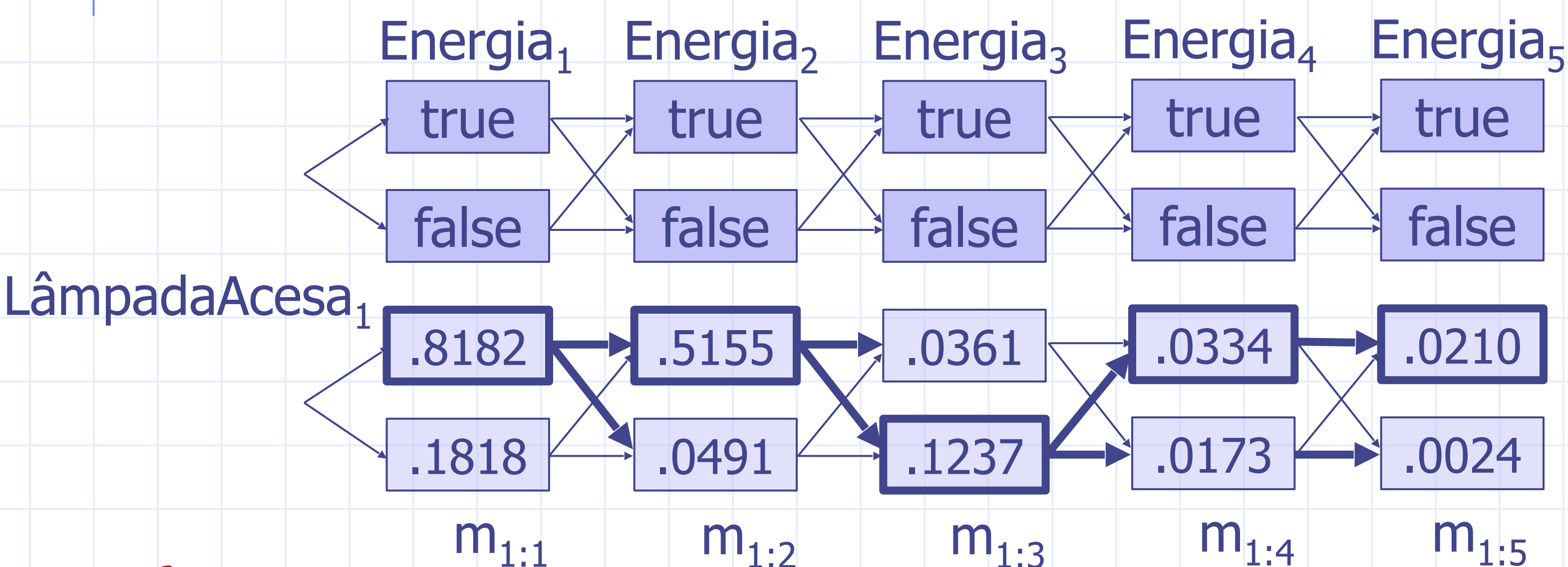
- Tarefa consiste em achar a seqüência de estados mais provável para uma combinação qualquer de variáveis de estado;
  - Ex.: Qual a seqüência de estados Energia, dado que a seqüência de observações LâmpadaAcesa é [true, true, false, true, true]?
- Uso do algoritmo de Smoothing para descobrir a distribuição posterior da variável Energia, a cada fatia de tempo;
- Em seguida, constrói-se a seqüência usando a cada passo o valor de Energia mais provável de acordo com a distribuição encontrada;

Inferência em Modelos Temporais

Seqüência Mais Provável

- Cada seqüência pode ser vista como um caminho em um grafo cujos nós são os estados possíveis a cada passo;
- Há uma relação recursiva entre os caminhos mais prováveis para cada estado x_{t+1} e os caminhos mais prováveis para cada estado x_t;

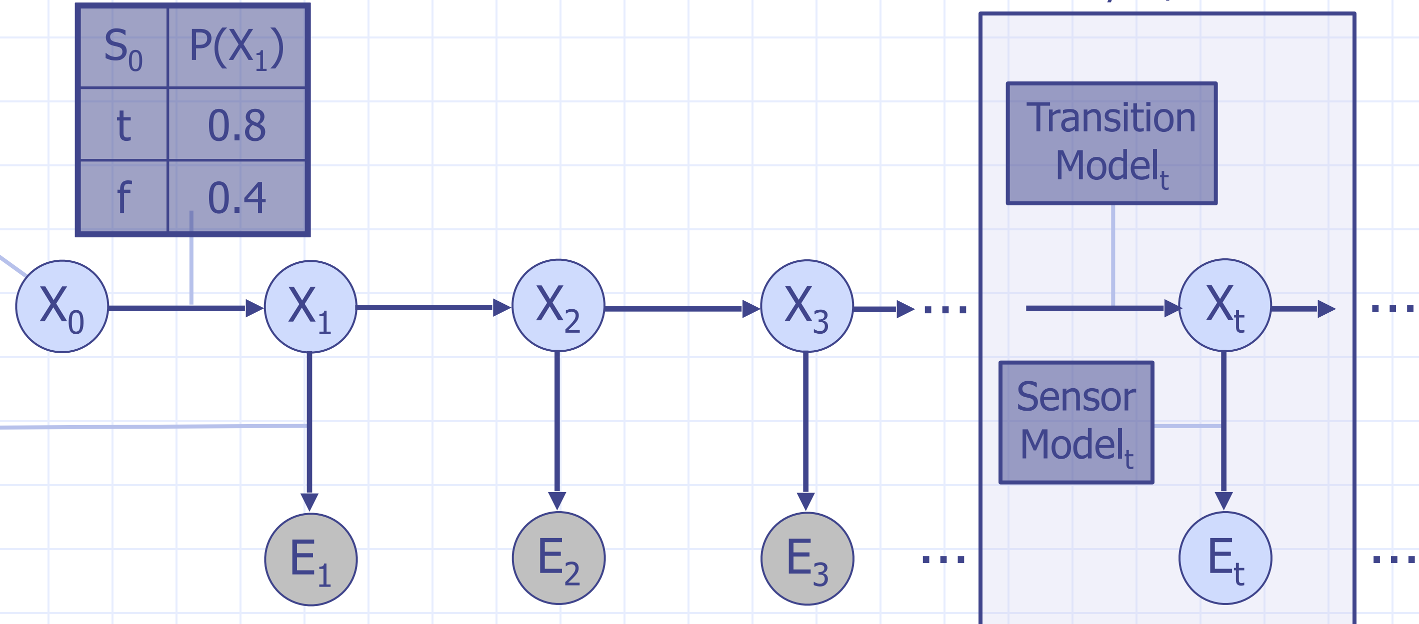
$$m_{1:t} = \max P(x_1, \ldots, x_{t-1}, X_t | e_{1:t}), \text{ para } x_1 \ldots x_{t-1}$$



Dynamic Bayesian Networks :: DBNs

Definição

- DBN é uma rede bayesiana que representa um modelo de probabilidades temporal.



Prior Distribution

$P(X_0)$
0.7

Transition Model

S_0	$P(X_1)$
t	0.8
f	0.4

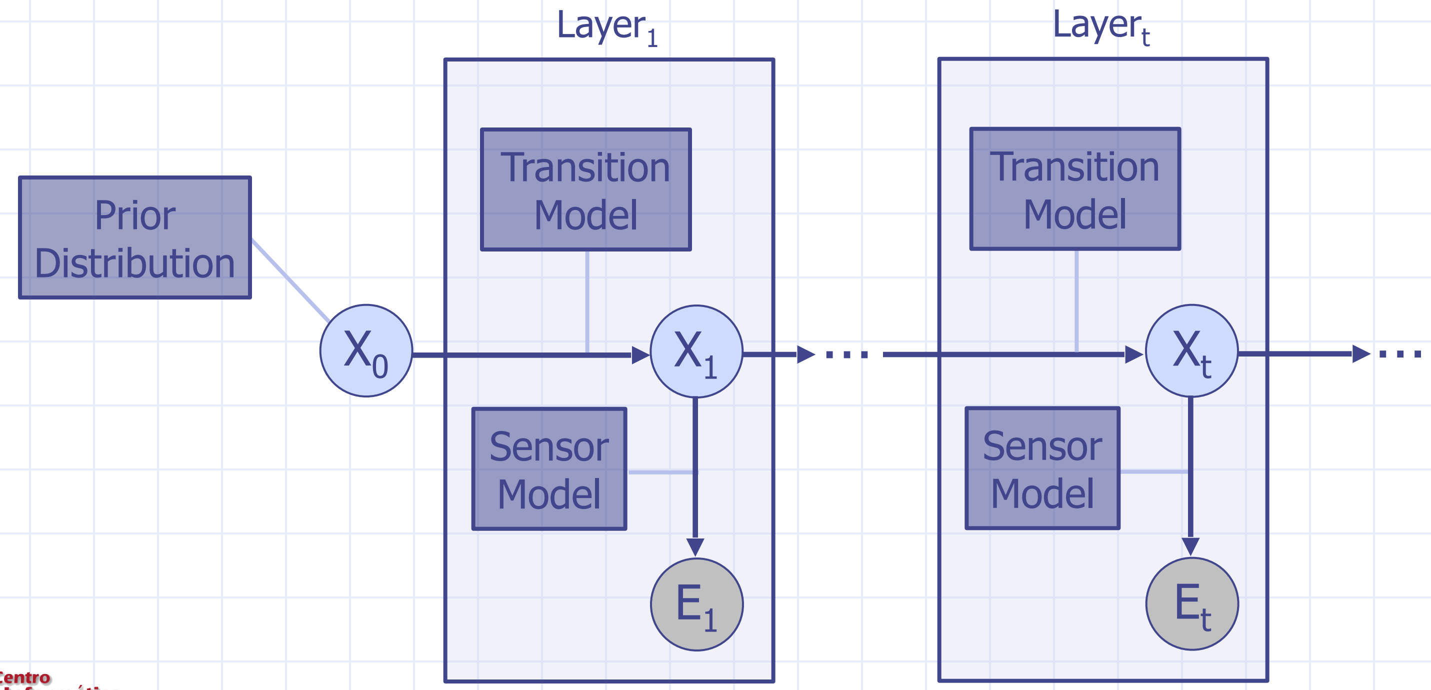
Sensor Model

X_1	$P(E_1)$
t	0.9
f	0.2

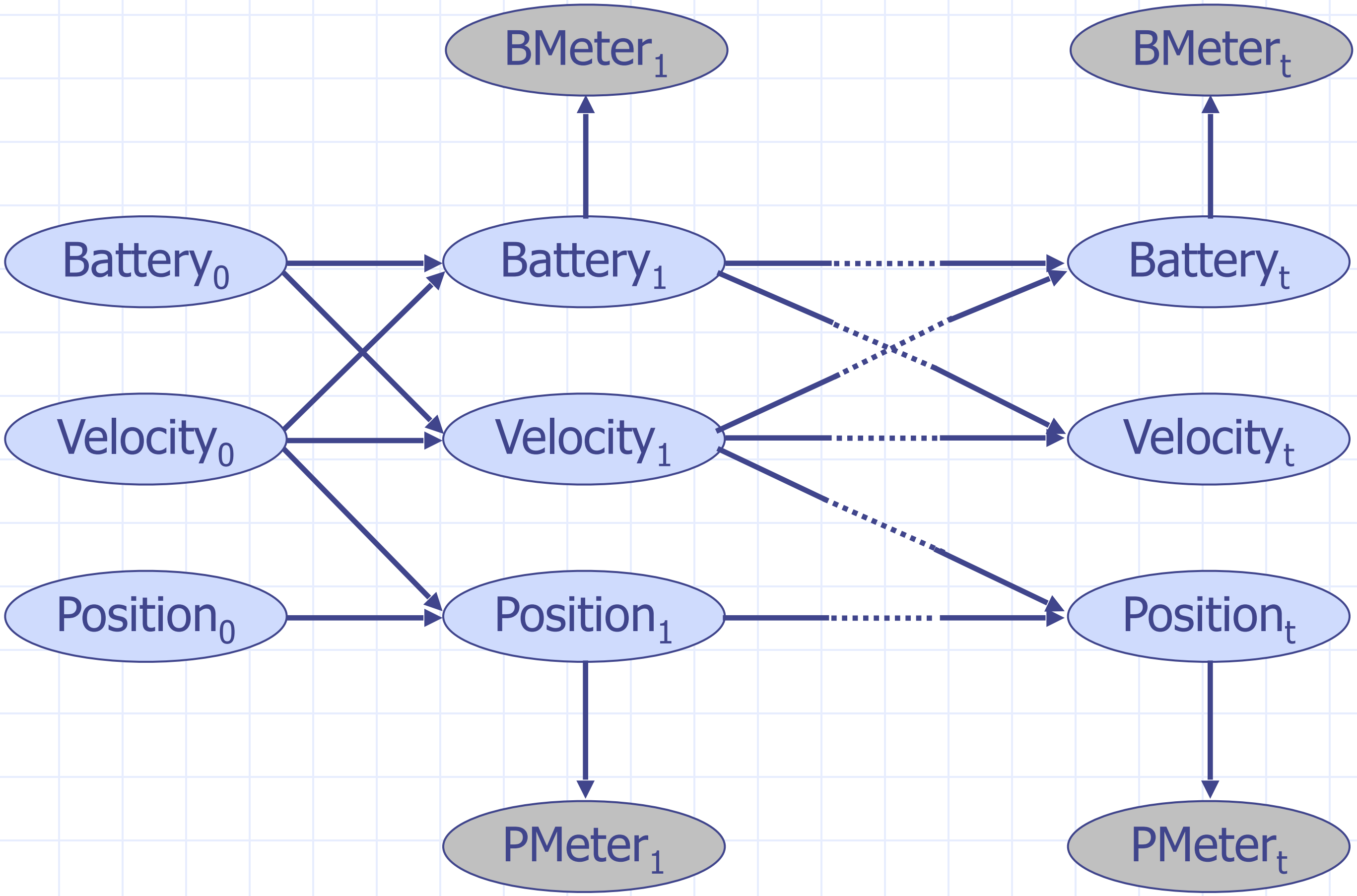
Dynamic Bayesian Networks :: DBNs

Premissas Adotadas

- Ligações entre variáveis são replicadas em cada camada;
- A DBN representa um processo de Markov de 1ª ordem;



Exemplo :: O Caso do Robô



Apêndice B – Falhas nas DBNs

Inferência Exata em DBNs

- Para um conjunto de observações, pode-se construir uma representação completa da DBN usando uma BN
  - Desdobramento da DBN até que a rede fique grande o suficiente para acomodar as observações
- Uma vez desdobrada, pode-se usar qualquer algoritmo de inferência em redes bayesianas
- Entretanto…

Inferência Exata em DBNs

- A aplicação de tais algoritmos não é eficiente!!!

- Filtering ou smoothing em uma seqüência longa de observações $e_{1:t}$
  - Espaco: O(t) crescendo sem limites à medida em que mais observações fossem adicionadas;
  - Tempo de inferência por atualização: crescente em O(t)

Inferência Exata em DBNs

- Voltando um pouco…
  - Usando recursão conseguimos tempo e espaço constantes na atualização do filtering
- Usando o algoritmo VARIABLE-ELIMINATION com as variáveis na ordem temporal é o mesmo que o filtering recursivo
- OK! Agora as más notícias…

Inferência Exata em DBNs

- Descobriu-se que o tempo e espaço "constantes" por atualização, na maioria dos casos, é exponencial de acordo com o número de variáveis de estado;

- Apesar dos custos serem menores que as atualizações dos HMM, eles ainda são inviáveis para grandes quantidades de variáveis;

Inferência Exata em DBNs

- Podemos usar DBNs para representar processos temporais muito complexos com várias variáveis esparçamente conectadas, mas **não podemos raciocinar de maneira eficiente e exata** sobre tais processos;
- Atualmente não há solução para este problema, embora muitas áreas da ciência e engenharia seriam beneficiadas enormemente por isso;
- Então…

Inferência Aproximada nas DBNs
Adaptação de Algoritmos de Inferência em BNs

- Algoritmo de aproximação que melhor se adapta ao contexto das DBNs:
  - Likelihood Weightening

- Entretanto, várias melhorias são necessárias;

- Voltando um pouco…

Inferência Aproximada nas DBNs
Likelihood Weightening Puro :: Problemas

- Likelihood Weightening puro:
  - Gera samples dos nós de estado da rede em ordem topológica, dando pesos a cada sample de acordo com a probabilidade de ocorrência, dadas as variáveis de evidência observadas

- Aplicar Likelihood Weightening diretamente na DBN desdobrada apresentaria os mesmos problemas de tempo e espaço.

Inferência Aproximada nas DBNs
Likelihood Weightening :: Adaptação

- Principal problema:
  - Cada sample é rodado por toda a rede, em turnos;
  - Samples são rodados em série;

- Proposta:
  - O conjunto inteiro de samples roda, uma camada por vez, por toda a rede;
  - Samples são rodados em paralelo;

Inferência Aproximada nas DBNs
Likelihood Weightening Puro :: Adaptação

- O algoritmo modificado pode ser visto como um algoritmo de Filtering:
  - conjunto de N samples corresponde à mensagem propagada (forward message)
- Inovações-chave do algoritmo:

1. Usa os próprios samples como uma representação aproximada da distribuição atual
2. Foca o conjunto de samples nas regiões de alta-probabilidade do espaço de estados

Inferência Aproximada nas DBNs

Particle Filtering :: Funcionamento

function PARTICLE-FILTERING(**e**, *N*, *dbn*) **returns** a set of samples for the next time step

inputs: **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 $\mathbf{P}(\mathbf{X}_1|\mathbf{X}_0)$, and sensor model $\mathbf{P}(\mathbf{E}_1|\mathbf{X}_1)$

static: *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 each time step t **do**

  for i = 1 **to** *N* **do**

    $S[i] \leftarrow$ sample from $\mathbf{P}(\mathbf{X}_1|\mathbf{X}_0 = S[i])$

    $W[i] \leftarrow \mathbf{P}(\mathbf{e}|\mathbf{X}_1 = S[i])$

  $S \leftarrow$ WEIGHTED-SAMPLE-WITH-REPLACEMENT(*N*, *S*, *W*)

return *S*

Inferência Aproximada nas DBNs
Particle Filtering :: Funcionamento

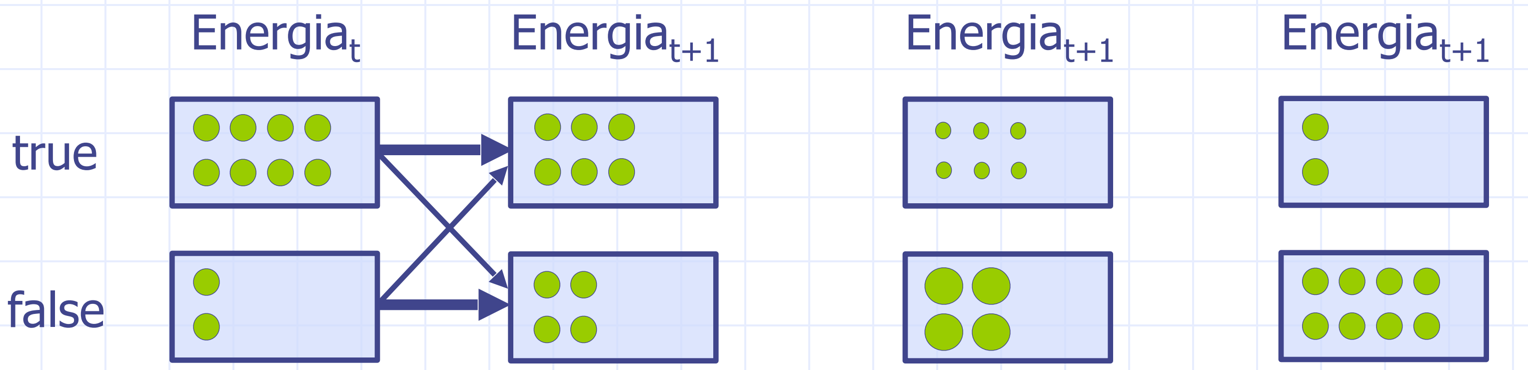
- Diagrama de Classes UML/OCL
- Diagrama de Atividades UML

Inferência Aproximada nas DBNs

Particle Filtering :: Exemplo de um ciclo de atualização

$N = 10$
$e_{t+1} = \neg$LâmpadaAcesa

O conjunto de samples vem de um ciclo prévio de atualização, ou, no caso de t=0, é gerado de acordo com a Prior Distribution.



(a) Propagate (b) Weight (c) Resample

$S \leftarrow$ WEIGHTED-SAMPLE-WITH-REPLACEMENT(N, S, W)

Inferência Aproximada nas DBNs
Particle Filtering

- Pode-se mostrar que o algoritmo é consistente:
  - Fornece as probabilidades corretas à medida em que N tende a infinito;
- Entretanto, embora o algoritmo pareça manter uma boa aproximação da realidade com um número constante de samples, não há garantias teóricas disso.

Inferência Aproximada nas DBNs
Particle Filtering :: Futuro e Aplicações

- Particle Filtering é atualmente uma área de estudos intensivos;
- Por ser um algoritmo de sampling, pode ser usado facilmente com DBNs hibridas e contínuas;
- Tal característica, permite sua aplicação em áreas como:
  - Padrões de movimentação complexos em video (Motion Tracking);
  - Previsão do mercado financeiro;
  - Reconhecimento de Voz

Raciocínio Probabilistico

Métodos de Computação Inteligente 2006.1

Ricardo Scholz e Vitor Dantas
<reps, vcd>@cin.ufpe.br

Apêncice A

Modelos de Markov Escondidos e Kalman Filters

Modelos de Markov Escondidos
Hidden Markov Models :: HMM

- É um modelo probabilistico temporal onde o estado do processo é descrito por uma única variável randômica discreta;

- Os valores possíveis da variável são os estados possíveis do mundo;

- Variáveis de estado podem ser adicionadas ao modelo temporal mantendo-se o framework da HMM:
  - Todas as variáveis de estado devem ser combinadas em uma "megavariável" cujos valores são todas as possíveis tuplas de valores das variáveis de estado individuais;

Modelos de Markov Escondidos

Desvantagens

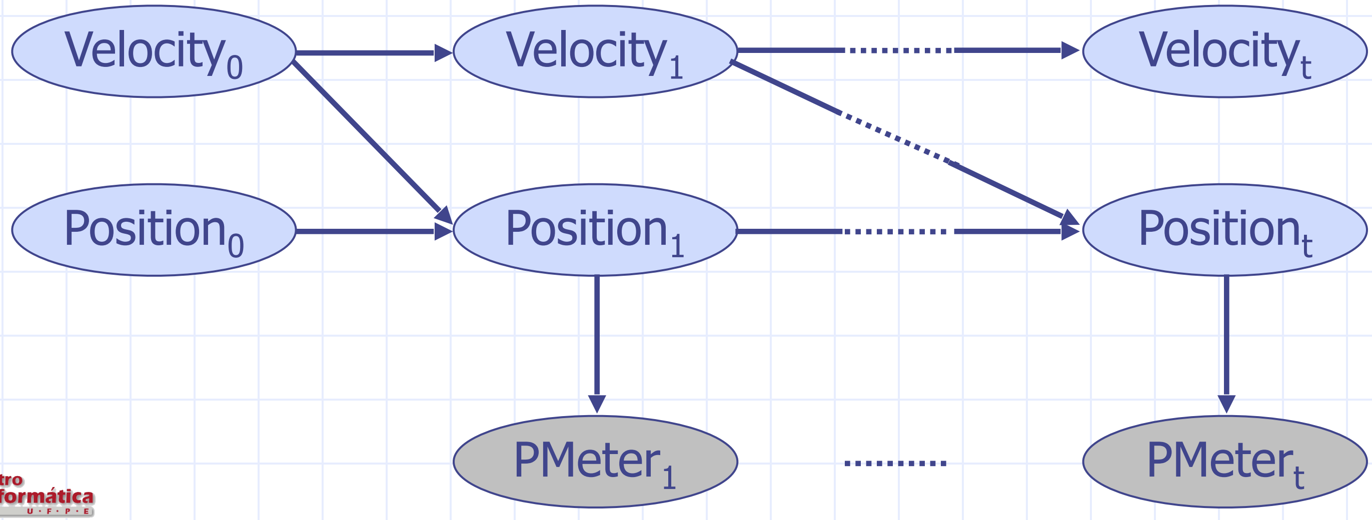
- Algoritmos são baseados em matrizes
  - Estrutura restrita permite uma implementação muito simples e elegante de todos os algoritmos básicos;
- Maior restrição dos algoritmos:
  - Requerem que a matriz de transição seja inversível;
  - Que modelo de sensores não tenham zeros:
    - ou seja, toda observação é possível em todos os estados;

Filtros de Kalman
Kalman Filters

- Motivação:
  - Estimar o estado de um sistema físico a partir de observações ruidosas ao longo do tempo;
- Problema pode ser formulado como inferência num modelo de probabilidade temporal:
  - o modelo de transição descreve a física do problema;
  - o modelo de sensores descreve o processo de medição;
- Representação algoritmos de inferência especiais para resolver estes problemas

Filtros de Kalman
Exemplo

- Tracking do vôo de um pássaro:
  - Variáveis de estado: S_t e V_t (posição e velocidade)
- Usando um modelo de transição gaussiano linear (para introduzir ruido), temos:
  - $P(S_{t+\Delta} = s_{t+\Delta} | S_t = s_t, V_t = v_t) = N(s_t + v_t\Delta, \sigma)(s_{t+\Delta})$



Filtros de Kalman
Aplicações e Análise

- Tracking de mísseis e aeronaves;
- Correntes oceânicas a partir de medições por satélite;
- Qualquer sistema caracterizado por variáveis de estado contínuas e medições ruidosas;

- Premissas são muito fortes
  - Modelos de transição e sensores gaussianos lineares;

Filtros de Kalman
Aplicações e Análise

- Há versões extendidas que permitem lidar melhor com não-linearidades
  - Extended Kalman Filters
    - Funciona bem para sistemas bem comportados
    - Permite atualizar a distribuição de estado gaussiana com uma aproximação razoável da verdadeira probabilidade posterior;
  - Switching Kalman Filters
    - Multiplos filtros de Kalman são executados simultaneamente, cada um utilizando um modelo diferente do sistema;

Centro de Informática U·F·P·E

Voltar

Apêncice B

Falhas Temporárias e Permanentes nas DBNs

Falhas nas DBNs

"Sensores de verdade falham..."

"...e eles não avisam que falharam!!!"

Falhas nas DBNs

- Fato: ruidos/falhas sempre acontecem nas medições;
- Tipos de falha:
  - Falha Temporária (Transient Failure)
  - Falha Persistente (Persistent Failure)
- Para um sistema lidar adequadamente com falhas de sensores, o modelo de sensores precisa incluir a possibilidade de falha!!!

Falhas Temporárias nas DBNs
Modelo de Erro Gaussiano

- Uso de distribuição gaussiana com variância pequena;
- Entretanto, a distribuição gaussiana é problemática:
  - Atribui probabilidades não nulas a valores negativos;
  - Para variáveis com limites restritos, a distribuição beta é mais apropriada.

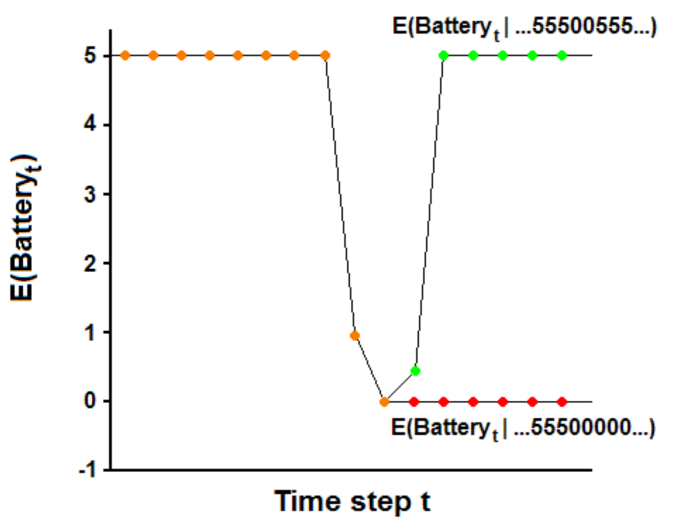
Falhas Temporárias nas DBNs
Transient Failure Model

- Solução: atribuição de probabilidades às falhas

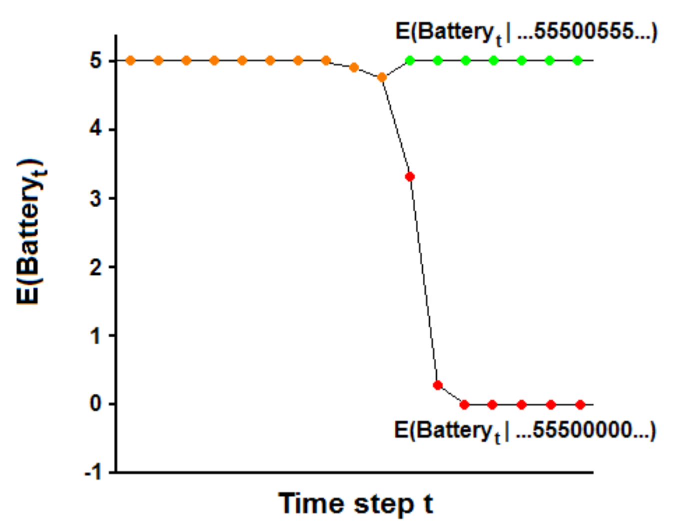
$$P(BMeter_t = 0 | Battery_t = 5) = 0.03$$

[Maior que a probabilidade dada pelo modelo de Erro Gaussiano]

Gaussian Error Model



Transient Failure Model



TFMs lidam com falhas sem causar danos catastróficos nas crenças.

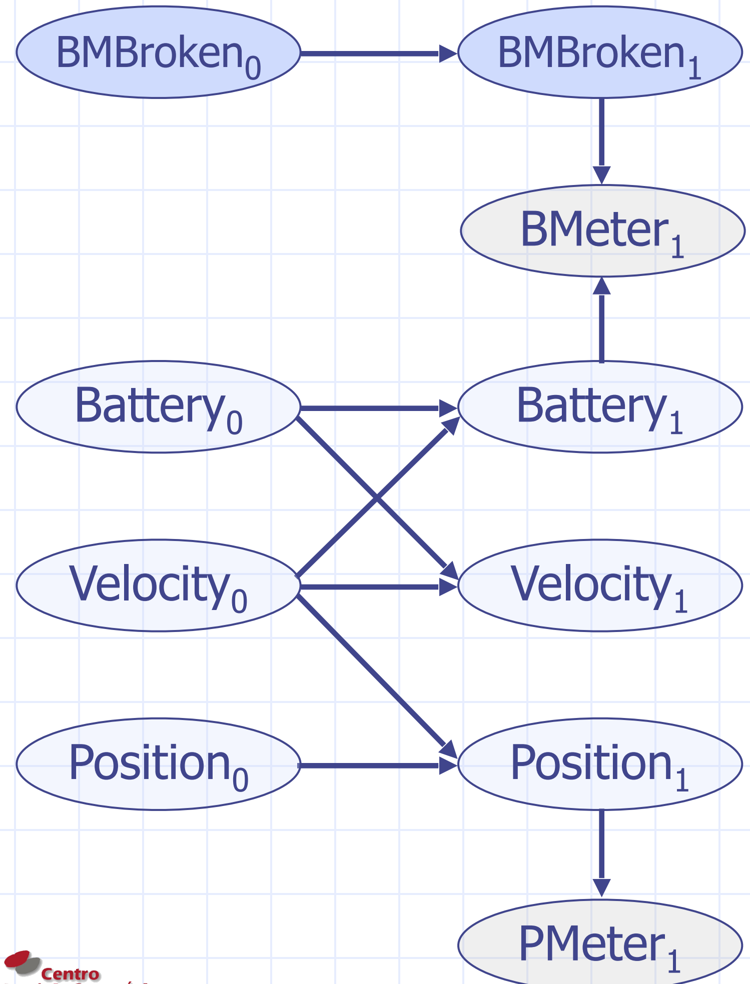
Falhas Persistentes nas DBNs

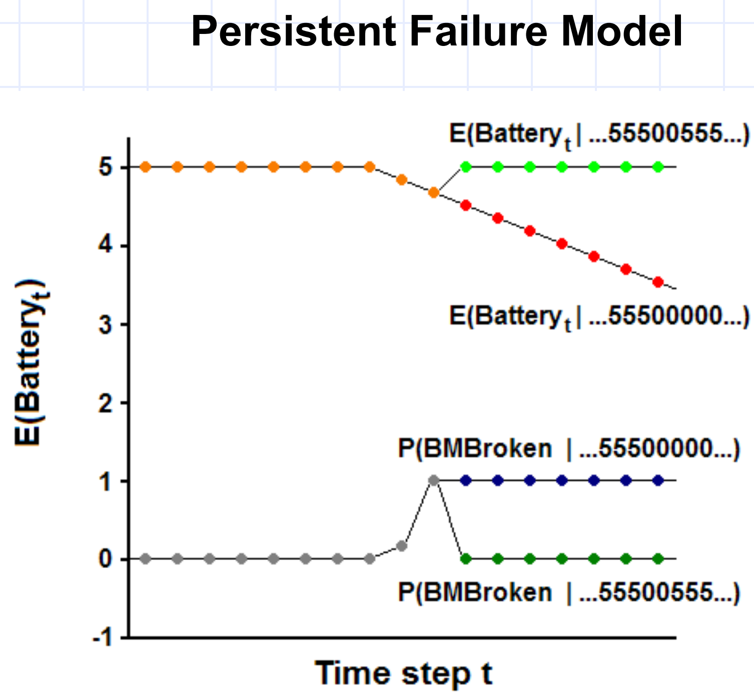
Persistent Failure Model

- Descreve como o sensor se comporta sob condições normais e após falha;
- Para isso, precisamos:
  - Inserir uma variável adicional (BMBroken) que descreve o status do sensor;
  - Representar falha persistente por um arco entre $BMBroken_t$ e $BMBroken_{t+1}$;
  - Atribuir uma pequena probabilidade de falha em cada fatia de tempo (ex.: 0.001);
  - Especificar que o sensor, uma vez quebrado, permanece neste estado para sempre, independente do estado real da bateria.

Falhas Persistentes nas DBNs

Persistent Failure Model





Voltar

Raciocínio Probabilistico

Métodos de Computação Inteligente 2006.1

Ricardo Scholz e Vitor Dantas
<reps, vcd>@cin.ufpe.br