RN, Chapter 17.4 — 17.5

Partially-Observable MDPs

Decision Theoretic Agents

- Introduction to Probability [Ch13]
- Belief networks [Ch14]
- Dynamic Belief Networks [Ch15]
- Single Decision [Ch16]
- Sequential Decisions [Ch17]
  - MDPs [Ch17.1 – 17.3]

    (Value Iteration, Policy Iteration, TD(λ))
  - POMDPs [Ch17.4 – 17.5]

    Dynamic Decision Networks
- Game Theory [Ch17.6 – 17.7]

Partially Accessible Environment

- In inaccessible environment

  percept NOT enough to determine state

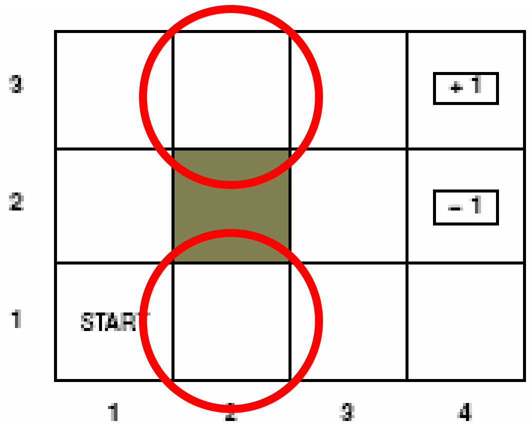
  Partially Observable

  Markov Decision Problem "POMDP"

$\Rightarrow$ Need to base decision on

DISTRIBUTION over possible states,

based all previous percepts, . . . (E)



Eg: Given only distance to walls in 4 directions,

"[2, 1] ≡ [2, 3]"

but DIFFERENT actions for each!

If P(Loc[2,1] | E) = 0.8, P(Loc[2,3] | E) = 0.2

then utility of action a is

$0.8 \times U(a \mid Loc[2,1]) + 0.2 \times U(a \mid Loc[2,3])$

Dealing with POMDPs

- Why not view “percept == state”...
  and just apply MDP alg to “percept”??

1. Markov property does NOT hold for percepts
   (percept ≠ states)
   - MDP means
     next state depends only on current state
   - But in POMDP:
     next percept does NOT depend only on current percept
2. May need to take action to *reduce uncertainty*
   . . . not needed in MDP, as always KNOW state
   ⇒ utility should include ValueOfInfo. . .

Extreme Case: Senseless Agent

- What if **NO** observations?
- Perhaps
  - act to reduce uncertainty
  - then go to goal

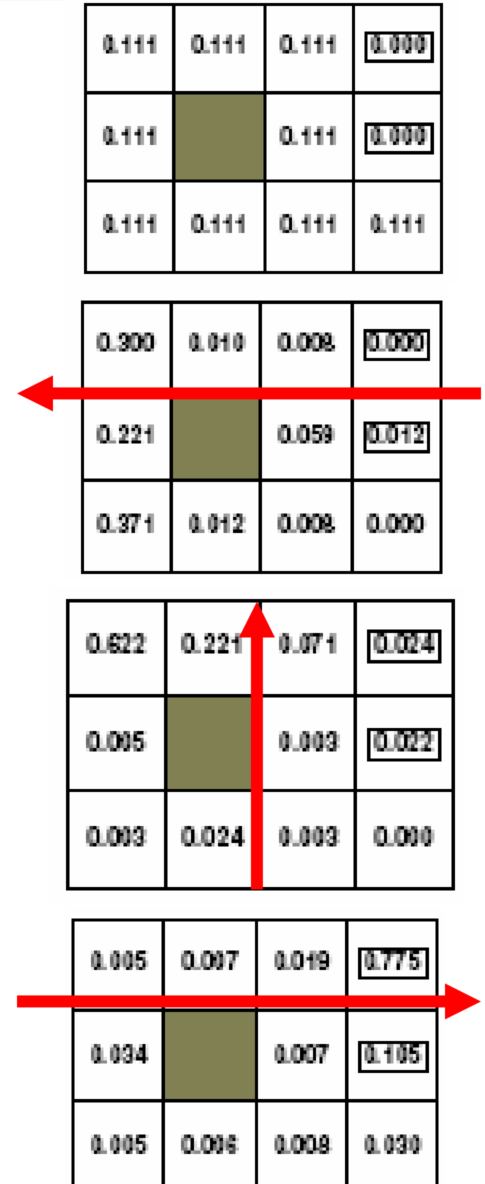
  (a) Initially: could be ANYWHERE

  (b) After "Left" 5 times

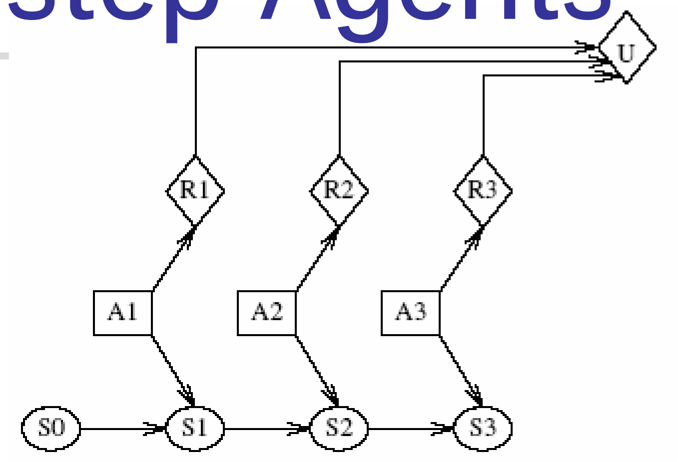
  (c) … then "Up" 5 times

  (d) … then "Right" 5 times
- Prob of reaching [4,3]: 77.5%

  but slow: Utility ≈ 0.08



“Senseless” Multi-step Agents



- Want **sequence of actions** $[a_1, \ldots, a_n]$
  that maximizes the expected utility:

  $\text{argmax}_{[a_1,\ldots,a_n]} \sum_{[s_0, \ldots, s_n]} P(s_0, \ldots, s_n \mid a_1, \ldots, a_n) \times U([s_0, a_1, \ldots , a_n, s_n])$
- If deterministic,
  use problem solving techniques to “solve”
  - (finding optimal sequence)
- Stochastic $\Rightarrow$ don't know state. . .
  but deal w/ DISTRIBUTION OVER STATES

Unobservable Environments

- View Action-Sequence as BIG action

For each possible actions-sequence $[a_1, \ldots, a_n]$
  compute $P(S_0, S_1, \ldots, S_n | a_1, \ldots, a_n)$
  compute $U([s_0, a_1, \ldots, a_n, s_n])$
  compute $score = \sum_{[s_0,\ldots,s_n]} P(s_0, \ldots, s_n | a_1, \ldots, a_n) \cdot U([s_0, a_1, \ldots, a_n, s_n])$
Return action-sequence that gave maximum $score$

- As Markovian:
  - $P(S_0, S_1, \ldots , S_n \mid a_1, \ldots , a_n) =$
    $P(S_0)\ \ P(S_1 \mid S_0, a_1) \times P(S2 \mid S1, a2) \times \ldots \times P(S_n \mid S_{n-1}, a_n)$
  - $U([s_0, a_1, \ldots , a_n, s_n]) = \sum_t R(s_t)$

$\Rightarrow$ For each action sequence,
requires searching over all possible sequences of resulting states.

- If $P(S_{t+1} \mid S_t, A_{t+1})$ deterministic, can be solved using search…

- Next action must depends on Complete Sequence of Percepts, **o**
  - (That is all available to agent!)
- Compress **o** into “distribution over states”
  - $\mathbf{p} = [p_1, \ldots, p_n]$ where $p_i = P(\text{state} = i \mid \mathbf{o})$
- Given new percept o_t,
  $\mathbf{p}' = [\, P(\text{state} = i \mid \mathbf{o}, o_t) \,]$

POMDPs

0.111	0.111	0.111	$\boxed{0.000}$
0.111		0.111	$\boxed{0.000}$
0.111	0.111	0.111	0.111

$\mathbf{b_{init}}$

- Partially Observable Markov Decision Problem
  - $\mathbf{M^a_{s,s'}} \equiv P(\ s' \mid s,\ a\ )$: transition
  - $\mathbf{R(s)}$: reward function
  - $\mathbf{O(s, o)} \equiv P(\ o \mid s\ )$: observation model

    [If senseless: $O(s, \{\}) = 1.0$]
- Belief state $b(.)$ $\equiv$ distribution over states
  - $b(s) \equiv P(\ s \mid \ldots\ )$ is prob b assigns to s
  - Eg: $b_{init} = \langle\ 1/9, 1/9, \ldots\ 1/9, 0, 0\ \rangle$
- Given $b(.)$, after action a, observation o
  - $b'(s') = O(s', o) \sum_s P(\ s \mid a, s'\ )\ b(s)$
  - $b' = \text{Forward}(\ b, a, o\ )$

Filtering!

- Optimal action depends only on current belief state!

  . . . not on actual state

What to do, in POMDP?

$$b'(s') = \alpha\, O(s', o) \sum_s P(s' \mid a, s')\, b(s)$$

- Policy π maps BELIEF STATE b to ACTION a
  $\pi(b) = a$ $\pi: [0, 1]^n \mapsto \{\text{ North, East, South, West }\}$

- Given optimal policy π^*
  - 1. Given b_i compute/execute action $a_i = \pi(b_i)$
  - 2. Receive observation o_i
  - 3. Compute $b_{i+1} = \text{Forward}(b_i, a_i, o_i)$

- With MDPs, can just "reach" new state … no observations...
  With POMDPs, need to know observation o_i to determine b'
- Some POMDP actions may be
  - to reduce uncertainty
  - to gather information

- How to compute optimal π^* ?
  . . . perhaps make POMDP look like MDP?

Transform POMDP into MDP ?

- Every MDP needs
  - Transition M: State Action $\mapsto$ Distribution over State
  - Reward   R: State $\mapsto$ $\Re$

$\Rightarrow$ Given "belief state" b, need

- ρ(b) = (expected) reward for being in b
  = $\sum_s$ b(s) R(s)

- μ(b, a, b') = P(b' | b, a)

  … prob of reaching b' if take action a in b. . .

  Depends on observation o:
  - P(b' | a, b) = $\sum_o$ P(b' | o, a, b) P(o | a, b)
    = $\sum_o$ δ[b' = Forward(b, a, o)] P(o | a, b)
  - where δ[b' = Forward(b, a, o)] = 1 *iff* b' = Forward(b, a, o)
- Need DISTRIBUTION over observations . . .

Distribution over Observations

- $P(\,o\,|\,a, b\,)$

$$
\begin{aligned}
&= \sum_{s'} P(\,o\,|\,a, s', b\,)\, P(\,s'\,|\,a, b\,) \\
&= \sum_{s'} O(s', o)\, P(\,s'\,|\,a, b\,) \\
&= \sum_{s'} O(s', o) \sum_{s} P(\,s'\,|\,a, s\,)\, b(s)
\end{aligned}
$$

- So...

$$
\begin{aligned}
\mu^{a}_{b,b'} &= P(\,b'\,|\,a, b\,) \\
&= \sum_{o} P(\,b'\,|\,o, a, b\,)\, P(\,o\,|\,a, b\,) \\
&= \sum_{o} [\![\, b' = Forward(b, a, o) \,]\!] \sum_{s'} O(s', o) \sum_{s} P(\,s'\,|\,a, s\,)\, b(s)
\end{aligned}
$$

POMDP $\Rightarrow^{?}$ MDP ??

- $\mu^{a}_{b,b'}$ = P(b' | b, a)
  ρ(b) = (expected) reward
  ... define OBSERVABLE MDP!
  (Agent can always observe its beliefs!)
- Optimal policy for this MDP π^*(b)
  is optimal for POMDP
  Solving POMDP on physical state space
  $\equiv$
  solving MDP on corresponding BELIEF STATE SPACE!
- But. . . this MDP has continuous
  (and usually HIGH-Dimension)
  state space!
- Fortunately . . .

Transform POMDP into MDP

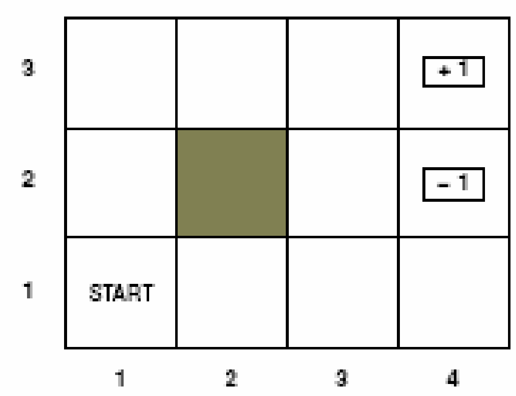
- Fortunately, ∃ versions of
  - value iteration
  - policy iteration

  that apply to such continuous-space MDPs

  (Represent $\pi(b)$ as set of REGIONS of belief space
    each with specific optimal action)

  U ≡ LINEAR FUNCTION of b w/in each region

  Each iteration refines boundaries of regions . . .



- Solution:

  [Left, Up, Up, Right, Up, Up, Right, Up, Up, ...]

   (Left ONCE to ensure NOT at [4,1],
   then go Right and Up until reaching [4, 3].)

  Succeeds 86.6%, quickly. . .

  Utility = 0.38

- In general: finding optimal policies is PSPACE-Hard!

Solving POMDP, in General

function DECISION-THEORETIC-AGENT(*percept*) **returns** *action*
calculate updated probabilities for current state based on
available evidence including current percept and previous action
calculate outcome probabilities for actions
given action descriptions and probabilities of current states
select *action* with highest expected utility
given probabilities of outcomes and utility information
return *action*

- To determine current state S_t:
  - Deterministic: previous action a_{t-1} from S_{t-1} determines S_t
  - Accessible: current percepts identify S_t
  - Partially accessible: use BOTH action and percepts
- Computing outcome probabilities:

  . . . as above
- Computing *expected utilities*:

  At time t, need to think about making decision D_{t+i}
  At that time $t+i$, agent will THEN have percepts $E_{t+1}, \ldots, E_{t+i}$
  But not known now (at time t). . .

Challenges

- To decide about A_t (action at time t), need distribution of current state based on
  - all evidence (E_i is evidence at time i)
  - all actions (A_i is action at time i)

$$\mathbf{\mathit{Bel}(S_t) \equiv P(\,S_t \mid E_1, \ldots, E_t, A_1, \ldots, A_{t-1})}$$

$\Rightarrow$ very hard to compute, in general

- But. . . some simplifications:
  - $P(\,S_t \mid S1, \ldots, St\text{-}1, A_1, \ldots, A_{t-1}) = P(S_t \mid S_{t-1}, A_{t-1})$

    Markov
  - $P(E_t \mid S_1, \ldots, S_t, E_1, \ldots, E_t, A_1, \ldots, A_{t-1}1\,) = P(E_t \mid S_t\,)$

    Evidence depends only on current world
  - $P(A_{t-1} \mid A_1, \ldots, A_{t-2}, E_1, \ldots, E_{t-1}) = P(A_{t-1} \mid E_1, \ldots, E_{t-1})$

    Agent acts based only input. . . and knows what it did

RECURSIVE form of *Bel()* updated with each evidence:

- **Prediction Phase**:

  Predict distribution over state, before evidence

$$\underline{\mathrm{Bel}}(S_t) = \sum_{s_{t-1}} P(S_t \mid S_{t-1} = s_{t-1}, A_{t-1}\,)\ \mathit{Bel}(S_{t-1} = s_{t-1})$$

- **Estimation Phase:** ... Incorporate E_t

$$\mathrm{Bel}(S_t) = \alpha\, P(E_t \mid S_t\,)\ \underline{\mathrm{Bel}}(S_t)$$

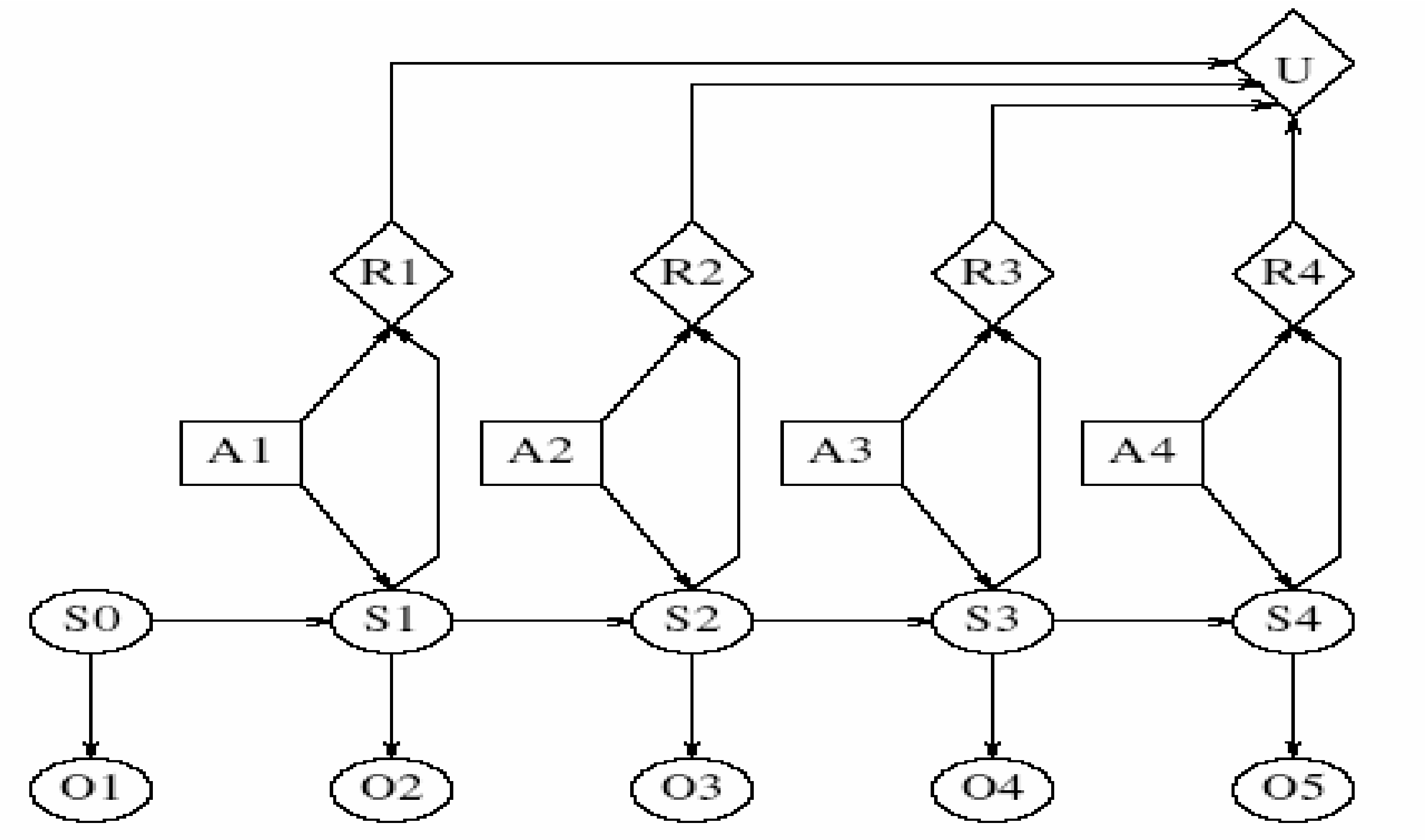
Decision-Theoretic Agent

function DECISION-THEORETIC-AGENT(E_t) **returns** an action
  inputs: E_t, the percept at time t
  static: BN, a belief network with nodes $\mathbf{X}$
    $Bel(\mathbf{X})$, a vector of probabilities, updated over time

  $\widehat{Bel}(\mathbf{X}_t) \leftarrow \sum_{\mathbf{x}_{t-1}} \mathbf{P}(\mathbf{X}_t \mid \mathbf{X}_{t-1}=\mathbf{x}_{t-1}, A_{t-1})\, Bel(\mathbf{X}_{t-1}=\mathbf{x}_{t-1})$
  $Bel(\mathbf{X}_t) \leftarrow \alpha\, \mathbf{P}(\mathbf{E}_t \mid \mathbf{P}\, X_t)\, \widehat{Bel}(\mathbf{X}_t)$
  $action \leftarrow \arg\max_{A_t} \sum_{\mathbf{x}_t} \left[Bel(\mathbf{X}_t=\mathbf{x}_t) \sum_{\mathbf{x}_{t+1}} P(\mathbf{X}_{t+1}=\mathbf{x}_{t+1} \mid \mathbf{X}_t=\mathbf{x}_t, A_t)\, U(\mathbf{x}_{t+1}) \right]$
  return *action*

- Dependencies are reasonable:
  - action mode: $P(S_t \mid S_{t-1}, A_{t-1})$
  - sensor model: $P(E_t \mid S_t)$

Partially Observable MDPs Dynamic Decision Networks



Approximate Method for Solving POMDP's

Two Key Ideas:

- Compute optimal value function $U(S)$ assuming complete observability
  (Whatever will be needed later, will be available)
- Maintain $Bel(S_t) = P(S_t | E_t, A_t, S_{t-1}, \ldots, S_0, E_0)$

At each time t

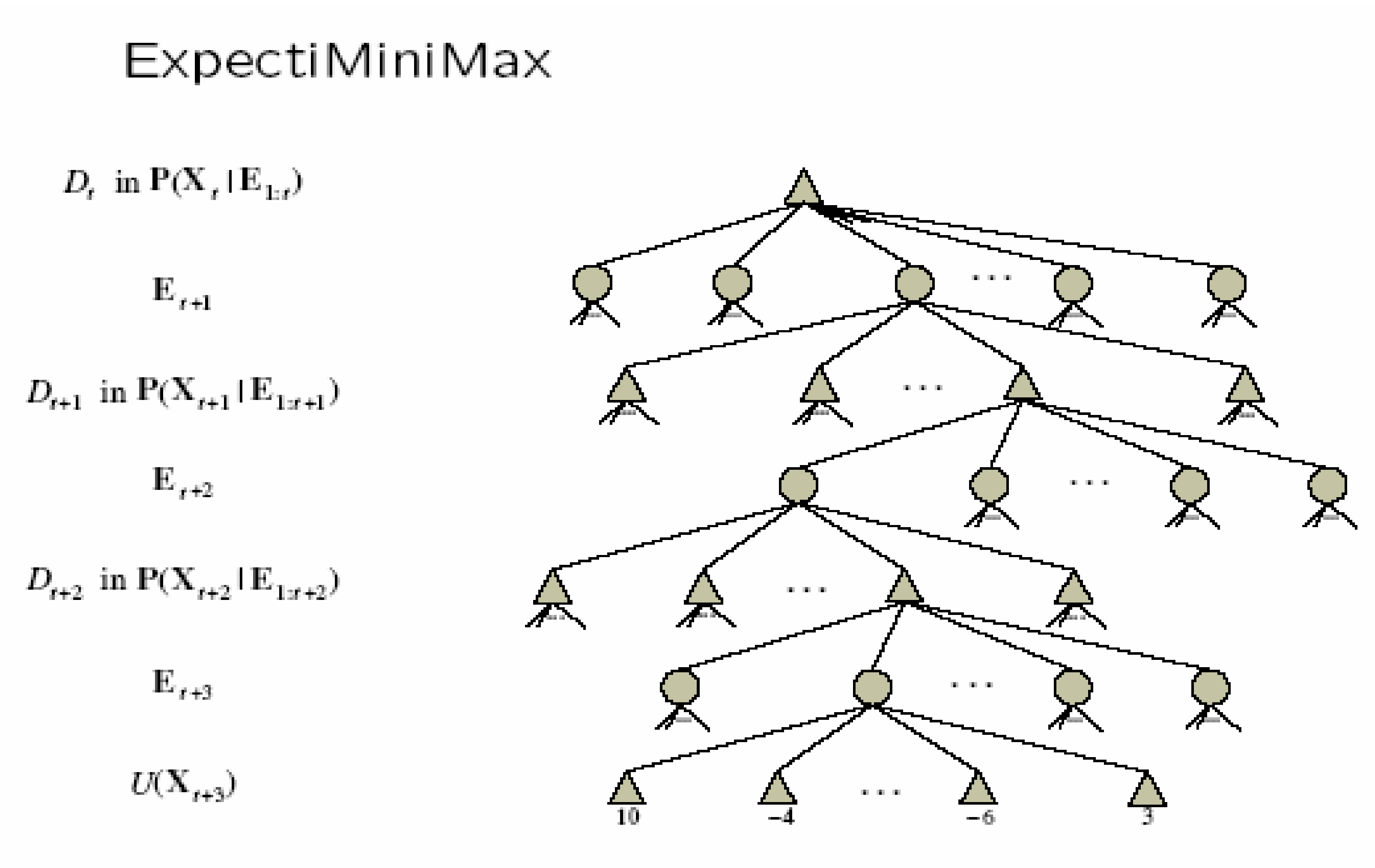
- Observe current percept E_t
- Update $Bel(S_t)$
- Choose next ***k*** optimal actions $[a_{t+1}, \ldots, a_{t+k}]$ to maximize

$$\sum_{S_{t+1},\ldots,S_{t+k}} \sum_{E_{t+1},\ldots,E_{t+k}} P(S_{t+1}|S_t, a_{t+1})]\ P(E_{t+1}|S_{t+1}) \cdots P(S_{t+k}|S_{t+k-1}, a_{t+k})$$

$$[\sum_{i=1}^{k} R(S_{t+i}|S_{t+i-1}, a_{t+i}) + U(S_{t+k})]$$

- Perform action a_{t+1}

Look-ahead Search



Wrt Dynamic Decision Networks

- Handle uncertainty correctly… sometimes efficiently…
- Deal with streams of sensor input
- Handle unexpected events (as have no fixed “plan”)
- Handle noisy sensors, sensor failure
- Act in order to obtain information

  as well as to receive rewards
- Handle relatively large state spaces

  as they decompose state into set of state var's

  with sparse connections
- Exhibit graceful degradation under time pressure

  and in complex environments using various approximation techniques

Open Problems wrt Probabilistic Agents

- First-order probabilistic representations

  If any car hits lamp post going over 30mph,
  occupants of car injured with probability 0.60.

- Methods for scaling up MDP's
- More efficient algorithms for POMDP's
- Learning environment
  - M^a_{ij}, $P(E \mid S)$, ...

Probabilistic Agents Summary

- **Three key components**:
  - P(S' | S,A)   (action model)
  - P(E | S)   (sensor model)
  - R(S' | S,A)   (reward function)
- In accessible environments,

  { Value iteration, Policy iteration } work well.

  Each computes local (state) utility function, optimal policy.
- In { unobservable, partially-observable } environments,

  lookahead search gives approx solutions
  - Updating current beliefs in a DDN is easy.
  - Look-ahead search is hard.