From HMMs to POMDPs to Bandits

Building up the equations from hypotheticals

Overview

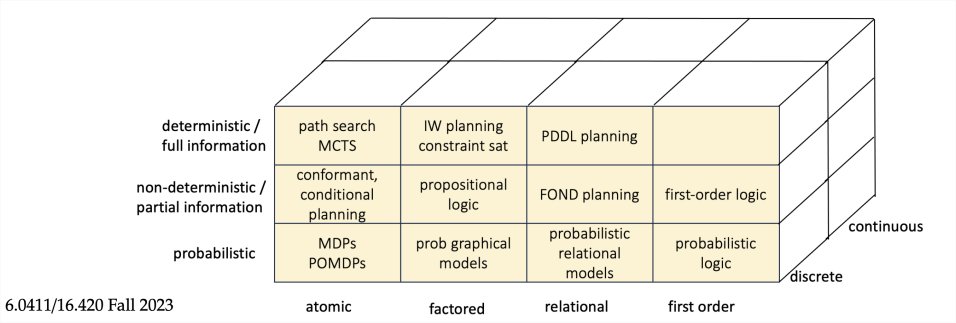


Figure 1: Summary of the algorithms covered in this class. We are discussing the bottom left tile (MDPs, POMDPS) today.

Color key

state s

action a

reward R

belief b

V value

transition T

observation likelihood

observation o

discount γ

Recap: HMM

Hidden states with:

- transition $T(s, s') = P(S_{t+1}{=}s' \mid S_t{=}s)$
- observation likelihood $P(O_t{=}o \mid S_t{=}s)$
- initial belief $b_0(s)$

No actions, no rewards. Passive process — you only observe.

HMM (review): forward algorithm (filtering)

$\alpha_t(s) = P(S_t{=}s,\ o_{1:t})$ — joint of current state and observations so far.

$$\alpha_t(s') \;=\; P(O_t{=}o_t \mid S_t{=}s') \sum_s T(s, s')\, \alpha_{t-1}(s)$$

- *Predict* with T, *correct* with the observation likelihood.
- Filtered belief: $b_t(s) = P(S_t{=}s \mid o_{1:t}) \propto \alpha_t(s)$.

HMM (review): backward algorithm (smoothing)

$\beta_t(s) = P(o_{t+1:H} \mid S_t{=}s)$: likelihood of *future* observations given state now.

$$\beta_t(s) = \sum_{s'} T(s, s')\, P(O_{t+1}{=}o_{t+1} \mid S_{t+1}{=}s')\, \beta_{t+1}(s'), \quad \beta_H(s){=}1$$

Combined forward–backward:

$$P(S_t{=}s \mid o_{1:H}) \;\propto\; \alpha_t(s)\, \beta_t(s).$$

What if we didn't observe anything?

No observations $\Rightarrow$ belief propagates under T alone:

$$b_t(s') \;=\; \sum_{s} T(s, s')\, b_{t-1}(s).$$

Why no forward/backward? Correct step becomes multiplication by 1, so α_t collapses to propagated prior. $\beta_t(s) = P(\text{empty} \mid s) = 1$ everywhere (rows of T sum to 1). Smoothed = filtered = propagated prior.

How can we calculate our expected reward

Suppose each state provides a reward $R(s)$ for occupying it at a particular time step. Using the relationship we derived on the previous slide, we can approximate the expected value associated with any given state. This value accounts both for the immediate reward received from being in that state at the current time step and for the expected rewards we are likely to obtain in future time steps as a consequence of being in that state now.

Fixed horizon H:

$$V_H(b_0) = \sum_{t=0}^{H-1} \sum_{s} b_t(s)\, R(s).$$

Infinite horizon, discounted by $\gamma \in [0, 1)$:

$$V(b_0) = \sum_{t=0}^{\infty} \gamma^t \sum_{s} b_t(s)\, R(s).$$

We can write this recursively

Fixed horizon H:

$$V_t(s) = R(s) + \sum_{s'} T(s, s')\, V_{t+1}(s')$$

with the base case $V_H(s) = 0$ for all states.

Infinite horizon, discounted by $\gamma \in [0, 1)$:

$$V(s) = R(s) + \gamma \sum_{s'} T(s, s')\, V(s')$$

Since the infinite case is a linear system we can solve it directly.

What if we have influence over the transition probabilities?

Add actions. Choosing a in s changes transitions to $T(s, a, s')$ and rewards to $R(s, a)$. Maximize expected reward.

For a fixed policy π:

$$V^{\pi}(s) \;=\; R(s, \pi(s)) \;+\; \gamma \sum_{s'} T(s, \pi(s), s')\, V^{\pi}(s').$$

Pick the best action at each state $\Rightarrow \max_a$.

Value Iteration

$$V^*(s) = \max_a \left[R(s,a) + \gamma \sum_{s'} T(s,a,s') V^*(s') \right]$$

Notice that if we only have one action, then $\max_a$ goes away, and we get the equation from slide 8. This is also the case if there are multiple actions but we can't control which one we pick, though the transition matrix would need to be updated to reflect this for the given step. For example, if we randomly pick an action with uniform probability, then we would form $T(s,s') = \frac{\sum_a T(s,a,s')}{|A|}$ where $|A|$ is the number of actions.

Value iteration

Since $V^*(s)$ isn't linear in general because of the max we generally can't solve it exactly, though we can approximate it recursively.

$$V_{k+1}(s) \leftarrow \max_a \Big[R(s,a) + \gamma \sum_{s'} T(s,a,s')\, V_k(s') \Big].$$

For $\gamma \in [0,1)$ it is guaranteed to converge. The expected lifetime for the γ is $\frac{1}{1-\gamma}$.

Hypothetical 3: actions and observations

Observations back. After acting, see o from $P(O_{t+1}{=}o \mid S_{t+1}{=}s', A_t{=}a)$. Only a belief.

Two pieces:

1. Belief update — HMM forward, action-conditioned.
2. Value function — defined on beliefs now.

What if we also make observations?

In the previous section, we worked our way up to value iteration by examining how to estimate future rewards when we cannot make any observations, but we do know our starting state (or a belief distribution over initial states). However, real life is more complicated. Frequently, we do not have perfect knowledge of our exact state and instead receive observations that update our beliefs about where we are and what future rewards we can expect from different actions. POMDPs model this kind of uncertainty.

What if we also make observations?

Recall that the HMM forward model is represented by the equation:

$$\alpha_t(s') \;=\; P(O_t{=}o_t \mid S_t{=}s') \sum_s T(s, s')\, \alpha_{t-1}(s)$$

We can write an analogous expression for the POMDP $b_{a,o}$: the new belief after action a, observation o.

$$b_{a,o}(s') \;=\; \frac{1}{\eta} P(O_{t+1}{=}o \mid S_{t+1}{=}s', A_t{=}a) \sum_s T(s, a, s')\, b(s)$$

This expression is identical to HMM forward step, only now T and observation likelihood are conditioned on the action.

What if we also make observations?

BUT WAIT! What about the backward step β?

In a POMDP, the goal is to estimate the underlying state accurately enough to choose the best possible action at the current time. Since the future is not observable, we can only run the backward pass starting from the current time step. Because β is initialized to 1 at that point, it does not affect the result and can simply be omitted.

What if we also make observations?

We have the belief update; now we need the value function. Previously V^* was indexed by s, but when the state is hidden the most we know is the belief b. So the value function must be defined on beliefs: $V^*(b)$.

The Bellman structure still holds, but we must rewrite each piece in terms of b rather than s.

What if we also make observations?

Rewrite each term of the MDP Bellman

$$V^*(s) = \max_a \Big[R(s,a) + \gamma \sum_{s'} T(s,a,s')\, V^*(s') \Big]$$

in terms of b.

Immediate reward. s is unknown; take its expectation under b:

$$R(s,a) \;\longrightarrow\; \sum_s b(s)\, R(s,a).$$

Future value. s' is unknown too, but each observation o updates the belief to $b_{a,o}$ via the POMDP forward step. Average the future value over observations we might receive:

$$\sum_{s'} T(s,a,s')\, V^*(s') \;\longrightarrow\; \sum_o P(o \mid b, a)\, V^*(b_{a,o}).$$

The $\max_a$ is unchanged.

POMDP Bellman equation

Putting the pieces together:

$$V^*(b) = \max_a \Big[\underbrace{\sum_s b(s)\, R(s,a)}_{\text{expected immediate reward}} + \gamma \sum_o P(o \mid b, a)\, V^*(b_{a,o}) \Big]$$

where

$$P(o \mid b, a) \;=\; \sum_{s'} P(o \mid s', a) \sum_s T(s, a, s')\, b(s).$$

Reinforcement learning vs. POMDP

In MDPs/POMDPs we assumed T and R were given. In real RL, the agent doesn't start knowing them. Uncertainty about T and R is itself a belief.

Optimal behavior trades off *exploitation* (act well now) vs. *exploration* (sharpen the belief).

Bandits

A bandit has:

- one state (no transitions);
- K actions ("arms"), each with unknown reward distribution;
- pull an arm $\Rightarrow$ observe a reward.

Why "horizon-1"? No state transitions means your action this round doesn't change what situation you face next round — every pull is independent. The planning problem at each step has depth 1: pick an arm, get a reward, done. There's no future state to reason about, only future *beliefs*. The only reason one pull affects the next is that observing its reward updates your belief about that arm's payoff.

Bandit as a POMDP

Finite-state example: each arm pays 1 w.p. $p_i \in \{0.1, 0.5, 0.9\}$; uniform prior.

- States: $(p_1, \ldots, p_K) \in \{.1, .5, .9\}^K$ — unknown parameters.
- Actions: which arm to pull.
- Transitions: identity (parameters don't change).
- Observations: the $0/1$ reward.
- Belief: factors into K independent distributions (one per arm).

Regret and UCB

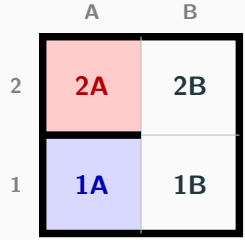
Regret after N pulls: reward of always playing the best arm, minus reward your policy got. Any policy has regret $\Omega(\log N)$.

UCB (KAlg-15). For each arm i, track $\hat{\mu}_i$ (empirical mean) and N_i (pulls of arm i); let $N =$ total pulls. Pull arm maximizing

$$\text{UCB}_i \;=\; \hat{\mu}_i \;+\; \sqrt{\frac{2\log(1+N\log^2 N)}{N_i}}.$$

- First term: exploit what we've learned.
- Second term: shrinks as $N_i \uparrow$, grows with N. Untried arm $\Rightarrow \infty$ bonus $\Rightarrow$ tried first.
- Achieves the optimal $O(\log N)$ regret.

Example: 2×2 grid world



Four states: 1A, 1B, 2A, 2B.

Walls: all outer edges, plus a wall between 1A and 2A. Hitting a wall means you stay put.

Special tiles:

- 1A — *sticky:* 50% chance of not moving.
- 2A — *trap:* $R=-1$; walls on N/S/W, so only East escapes (we are no longer assuming traps can not be left).

Actions N, S, E, W: 70% in the chosen direction, 10% move in each other direction. Blocked ⇒ stay.

Rewards: -1 in 2A, $+1$ elsewhere.

Note: This setup is **slightly different** than the one on the website (davidkoplow.github.io/searching_demos)

Example)

Action North (70%↑, 10% each S/E/W)

from\to	$1A$	$1B$	$2A$	$2B$
$1A$	0.95	0.05	0	0
$1B$	0.1	0.2	0	0.7
$2A$	0	0	0.9	0.1
$2B$	0	0.1	0.1	0.8

Action South (70%↓, 10% each N/E/W)

from\to	$1A$	$1B$	$2A$	$2B$
$1A$	0.95	0.05	0	0
$1B$	0.1	0.8	0	0.1
$2A$	0	0	0.9	0.1
$2B$	0	0.7	0.1	0.2

Example

Action East (70%→, 10% each N/S/W)

from\to	$1A$	$1B$	$2A$	$2B$
$1A$	0.65	0.35	0	0
$1B$	0.1	0.8	0	0.1
$2A$	0	0	0.3	0.7
$2B$	0	0.1	0.1	0.8

Action West (70%←, 10% each N/S/E)

from\to	$1A$	$1B$	$2A$	$2B$
$1A$	0.95	0.05	0	0
$1B$	0.7	0.2	0	0.1
$2A$	0	0	0.9	0.1
$2B$	0	0.1	0.7	0.2

Example

Using the grid and transition matrices above, with $\gamma = 0.9$ and $R = (+1, +1, -1, +1)$ for $(1A, 1B, 2A, 2B)$:

Q. Part 1 (passive). *The agent picks actions uniformly at random. Let* $T_{\text{rand}} = \frac{1}{4}(T_N + T_S + T_E + T_W)$. *Solve for* V *using value iteration.*

Q. Part 2 (MDP). *The agent sees its state. Run value iteration to find* V^* *and* π^*.

Q. Part 3 (POMDP). *The agent observes only* $o \in \{\text{Normal}, \text{Trap}\}$ *(deterministic: Trap iff in 2A). From* $b_0 = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$, *the agent sees Normal, transitions passively, then sees Normal again. What is the best next action?*