Approximate Policies for MDPs

- Till now we have seen some special MDPs for which we know the optimal policy! To summarize:
  - Monotone I-sla in optimal stopping
  - bang-bang / linear / threshold policies in lineardynamics +
    linear / quadratic / convex objectives
  - Index policies for multi-armed bandits

Unfortunately (well, fortunately if you work in controls...), most MDPs live outside these islands; this gets us to the ocean of approximate DP and the ‘curse of dimensionality’

- There are too many approaches to ADP to try and cover all. However, most are based on a few basic principles
  - comparing to static relaxations
  - coupling to dynamic relaxations
  - showing improvement in iterative schemes
  - focussing on ‘typical’ trajectories (‘good events’)

We will see some examples of each technique - warning: there are not always the ‘best’ result in each setting, but more chosen for their illustrative power
Technique 1 - Dual balancing for weakly coupled LPs

Consider first a classical problem from economics

- Pandora's Box (Weitzman '79)
  - Given m boxes, where box i = cost c_i, unknown value V_i ~ F_i
  - We can open any number of boxes but can select only one open box
  - Aim: Maximize reward of selected box - cost of searching

This is a special case of Gittins' index!

- Imagine at each 'time', we could open new box or select open box
- Let T be box opened in round t, V_t be discovered value, and T be the stopping time

$$E[\text{Reward}] = E[\sum_{t=0}^{T-1} \beta^t c_t + \beta^T \max \{V_0, V_1, \ldots, V_{T-1}\}]$$

- Boxes can be in states \{Unopened\} x \mathbb{R} = value of opened
  A box's state remains same if not opened/selected (i.e. its non-restless). If value x \to round x/(1-\beta) in future

- For single box i: \( V_i(x) = x(1-\beta) \)
  \( V_i(\text{Unopened}) = -c_i(1-\beta) + \beta E[\max(V_i, \text{Unopened})] \)

Let \( G_i = V_i/(1-\beta) \). As \( \beta \to 1 \), we get \( G_i(x) = x \)

\( G_i(\text{Unop}) = -c_i + E[\max(V_i, G_i(\text{Unop})]) \)

Opt strategy - Open boxes in order of \( G_i \) till max\{\( V_1 \), \( V_2 \)\} exceeds highest remaining \( G_i \)

- Ex: If \( V_i = \text{var}, \) stop 1-p; else 0 \( \Rightarrow \) \( G_i = V_i - c_i/\pi \)

- Now consider 2 variants which are not indexable
  - Have a fixed budget B for Pandora's box (budgeted search)
  - Boxes appear sequentially - must stop to select item (prophet inequality)

We will now see a technique that gives a good approximation to this.
Dual Balancing
(Based on notes by Kamesh Munagala)

- Consider the budgeted Pandora’s box problem with all $C_i = 1$ and budget $k$ (i.e., can open $\leq k$ boxes before selecting)
- We first want to upper bound the optimal reward via a relaxation
  - For any policy $\pi$, let $E_j = 1P[\text{Box } j \text{ is explored}]
    \quad S_j(o) = 1P[\text{Box } j \text{ is selected and } V_j = o]$
  
  Moreover let $\mathbb{E}[E_j] = e_j$, $\mathbb{E}[S_j(o)] = S_j(o)$
  
  - $\mathbb{E}[\text{Reward}(\pi)] = \sum_{j \in [m]} \sum_{o \in \mathcal{V}} e_j \cdot S_j(o)$

  Finite set of values - not important though

- Natural ‘floor’ constraints for $(E_j, S_j(o))$:
  
  - (only one selection) $\sum_j E_j \leq 1$  \quad \text{prob of } V_i = o
  
  - (only opened box is selected) $S_j(o) \leq E_j \cdot p_j(o)$ \quad \forall j \in [m], \forall o
  
  - (at most $k$ boxes opened) $\sum_{j \in [m]} e_j \leq k$

  $E_j, S_j(o) \geq 0$

This LP is a relaxation since it ignores correlations between $S_j(o), V_i$

- We can further relax this to get

$$P: \quad \max \sum_j \left( \sum_o o \cdot S_j(o) \right)$$

s.t. $\sum_j \left( \frac{e_j}{k} + \sum_o S_j(o) \right) \leq 2$  \quad \text{(coupling constraint)}$

$S_j(o) \leq e_j \cdot p_j(o) \quad \forall j, o$

(\text{per-box feasibility})

Moreover by Lagrangean duality, we have

$$L(\lambda): \quad \max \sum_j \left( \lambda e_j + \sum_o (\lambda - \lambda) S_j(o) \right)$$

s.t. $S_j(o) \leq e_j \cdot p_j(o) \quad \forall j, o$

and $L(\lambda) \geq P \geq \mathbb{E}[\text{Reward}(\pi)] \quad \forall \lambda > 0$, policy $\pi$
- Now given \( \lambda \), maximizing \( L(\lambda) \) is easy since boxes are decoupled! For each box \( j \in [m] \), we maximize
\[
\Phi_j(\lambda) = E[-E_j(\lambda/k) + \sum_0 S_j(\lambda) (v-\lambda) \]
(i.e., open at cost \( \lambda/k \), get value \( (v-\lambda) \) if \( V_j = 0 \) and select)
Thus \( \Phi_j(\lambda) = \max (0, E[(V_j-\lambda)^+] - \lambda/k) \)
don’t open \( \hat{\Pi}_j(\lambda) \) \( \text{open and select if } V_j \geq \lambda \)

We also define \( \hat{\Pi}_j(\lambda) \) to be this optimal policy, and denote \( E_j(\lambda) = E_j^*, S_j(\lambda) = \sum_0 s_j^*(\lambda), R_j(\lambda) = \sum_0 v s_j^*(\lambda) \) under \( \hat{\Pi}_j(\lambda) \)
Then
\[
L(\lambda) = 2\lambda + \sum_j \left(R_j(\lambda) - (\lambda/k)E_j(\lambda) - \lambda S_j(\lambda)\right)
= 2\lambda + \sum_j \Phi_j(\lambda)
= 2\lambda + \sum_j \left(E[(V_j-\lambda)^+] - \lambda/k\right)^+
\]
continuously increasing in \( \lambda \), \( \sum_j \Phi_j(\lambda) = \sum_j E[V_j] \)

- Assuming \( \sum_j E[V_j] > 0 \), we can choose \( \lambda^{bal} \) s.t.
\[
2\lambda^{bal} = \sum_j \Phi_j(\lambda^{bal}) \quad \text{(dual balancing)}
\]
Also let \( \text{OPT} = E[(\text{Reward}(\hat{\Pi}))] \) for the optimal policy.
Claim - \( \text{OPT} \leq 3\lambda^{bal} \)

- Finally we get our approximate policy \( \hat{\Pi} \)
1) Compute \( \lambda^{bal} \)
2) Consider boxes in arbitrary order
3) For each box, execute \( \hat{\Pi}_j^{*}(\lambda) \) (optimal single box policy)
   a) If you select a box, or run out of budget, stop

Then - \( E[(\text{Reward}(\hat{\Pi}))] \geq \lambda^{bal} \geq \text{OPT}/3 \)
Pf. - For each box \( j \), we have by defn
\[
R_j(x^m) = \Phi_j(x^m) + x^m S_j(x^m) + (x^m/2)E_j(x^m)
\]
This is equivalent to a problem where for each box \( j \), we get \( \Phi_j(x^m) \) if we ‘consider’ it, \( (x^m/2) \) if we explore it, and \( x^m \) if we select it.

Recall \( L(x^b) = x^b + x^m + \sum_j \Phi_j(x^m) \)
Now suppose we run policy \( \tilde{\pi} \). Due to budget constraints, we may not be able to consider every box. However, for every sample path, at least one of the following happen:
1) A box \( j \) is selected
2) We exhaust budget (i.e., explore \( 2 \) boxes)
3) We consider every box

Under the amortized reward scheme, 1) leads to reward \( x^m \) (by selection), 2) leads to reward \( \sum_{i=1}^2 x^m/2 \) (by exploration), and 3) leads to reward \( \sum_j \Phi_j(x^m) = x^m \)
\[\implies \mathbb{E}[\text{Reward}(\tilde{\pi})] = x^m > x^b \geq \text{OPT} / 3\]

Notes
- The above algorithm and analysis works for the general budgeted version by replacing exploration cost by \( c_l \lambda / k \)
- The guarantee is the same for known fixed arrival order, uniform random arrival order, arbitrary order on free order. It does not 'separate' these models
- The algorithm is fixed upfront (static) - does not adapt to state
- This is typically a harder problem
- The recourse ability did not matter for the guarantee
- The guarantee is actually against a stronger LP benchmark
The Prophet Inequality

Setting - Arrivals, with arrival \( t \) offering reward \( V_t \sim F_t \)
- Want to accept a single arrival (optimal stopping)

The Prophet Benchmark

The name of this problem comes from the fact that we now want to compare against an ex-post benchmark

\[
R(\text{prophet}) = \mathbb{E}[\max_{t \in T} V_t]
\]

This can be viewed as the solution achieved by a ‘prophet’ who knows all \( \{V_t\} \) before deciding which to accept. We can write this as the following LP

\[
\begin{align*}
\mathcal{P}_\text{pro}(V) : \max \sum_{t \in T} x_t V_t \\
\text{s.t. } \sum_{t \in T} x_t &\leq 1 \\
\quad x_t &\geq 0 \quad \forall t
\end{align*}
\]

\[
(\text{dual}) \min \theta \quad \text{s.t. } \theta &\geq V_t \forall t \\
\theta &\geq 0
\]

Note that this is unachievable. Indeed, if \( T = 2 \), \( V_1 = 1 \) and \( V_2 = 1/2 \beta_e(1/4) \), then \( \mathbb{E}[\mathcal{P}_\text{pro}(V)] = M \cdot 1/4 + 1 - 1/4 = 2 - 1/4M \), while an online policy either takes \( V_1 \) on \( V_2 \), and gets expected reward 1.

The Fluid Benchmark

Note the benchmark \( \mathcal{P}(V) \) is actually a random variable! For each sample path of \( V \), we get a different value. To get around this, we do a bait-and-switch by actually comparing against an even larger fluid benchmark

Let \( Z_t(\omega) = \mathbb{I}[\text{ALG chooses arrival } t \text{ AND } V_t = \omega] \)

\[
\begin{align*}
\mathcal{P}_\text{fluid} : \max \sum_{t \in T} \sum_{\omega \in T} V_t Z_t(\omega) \\
\text{s.t. } \sum_{t \in T} \sum_{\omega \in T} Z_t(\omega) &\leq 1 \\
Z_t(\omega) &\leq p_t(\omega) \forall t, \omega \\
Z_t(\omega) &\geq 0 \\
\end{align*}
\]

\[
\lambda(\omega) : \lambda + \max \sum_{t \in T} \sum_{\omega \in T} Z_t(\omega) (2\lambda) \\
\text{s.t. } Z_t(\omega) &\leq p_t(\omega) \forall t, \omega \\
Z_t(\omega) &\geq 0 \\
\]
As before, for any $x > 0$, $L(x) = P_{\text{fluid}} \geq E[P_{\text{Te}}(y)] \geq E[R_{\text{opt}}(y)]$
for any online policy $\pi$.
Again, we can look at a single-item problem

$$L_t(\lambda) = \max \sum_v z_t(v)(v-\lambda)$$

subject to

$$z_t(v) \in [0, p_t(v)] \forall v$$

Now, opt policy $\pi_t^*(x) = \mathbb{E}[v_t] = 1_{\{v_t > \lambda\}}$

$$L_t(\lambda) = E[L(V_t - \lambda)^+] = R_t(\lambda) - \lambda S_t(\lambda)$$

$$\sum_t z_t^*(v) \geq \sum_t z_t^*(v)$$

Similar to before, $R_t(\lambda) = L_t(\lambda) + \lambda S_t(\lambda)$, which we can reinterpret as an alternate amortized problem where we get $L_t(\lambda)$ if we 'consider' arrival $t$, and $\lambda$ if we 'select' arrival $t$.

Returning to the original problem, we again choose $\lambda^{bal} = \Sigma_t L_t(\lambda^{bal}) = \lambda^{bal} = \Sigma_t E[(V_t - \lambda^{bal})^+]$, and accept first arrival that exceeds $\lambda^{bal}$. Call this policy $\tilde{\pi}$.

By amortization argument, along each sample path, we either get $\Sigma_t L_t(\lambda^{bal})$ if we see all arrivals, or $\lambda$ if we select any earlier arrival $\Rightarrow E[R_{\text{opt}}(\tilde{\pi})] = L(\lambda^{bal})/2$.

Thus we get

$$E[R_{\text{opt}}(\tilde{\pi})] \geq \frac{1}{2} E[\max_{t \in E} (V_t)]$$

Where $\tilde{\pi} = \text{Accept first } V_t \text{ that satisfies } V_t \geq \lambda^{bal}$

More generally, if LP relaxation of MDP has $d$ coupling constraints, s.t. if we remove them, then we can solve the problem optimally, then dual balancing gives a policy which gets $E[R_{\text{opt}}] = \frac{1}{d} \text{ OPT}$.