Learning All Optimal Policies with Multiple Criteria

Leon Barrett & Srini Narayanan

University of California, Berkeley

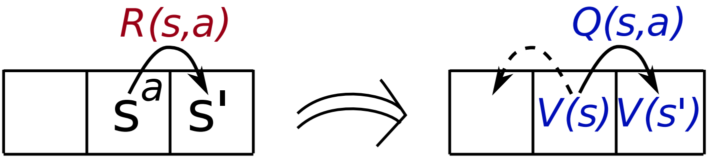
July 7, 2008

Motivation

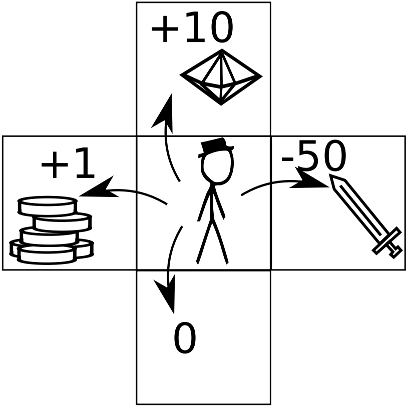
- Standard Reinforcement Learning: single reward
- Multi-criterion learning, reduce to standard RL
  - e.g. Natarajan and Tadepalli. *Dynamic Preferences in Multi-Criteria Reinforcement Learning.* Proc. ICML, 2005.
- We lift to solve over all preferences at once
  - Can view all optimal policies
  - Can change preferences at runtime, without relearning

Reinforcement Learning: Important Components

- Maxmimize expected discounted reward
  - Summarize with V and Q
- Bellman equations: recurrence
  - $Q^*(s,a) = \mathbb{E}[R(s,a) + \gamma V^*(s')]$
  - $V^*(s) = \max_a Q^*(s,a)$



Reward Decomposition

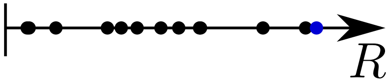


- Arbitrary choices
- Or twiddle to get desired behavior

- We make weights explicit: $R(s, a) = \overrightarrow{R}(s, a) \cdot \overrightarrow{w}$

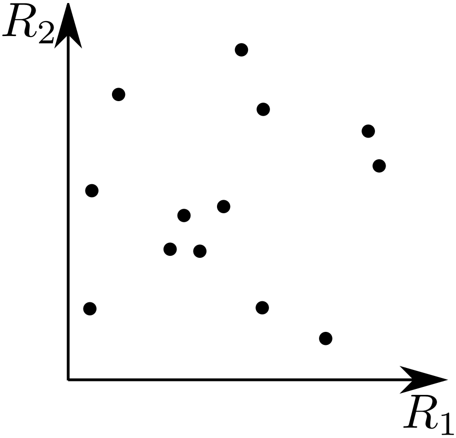
Q-Values in Space!

- $V(s_0)$
- $Q(s_0, a_0)$
- Each policy gives one value



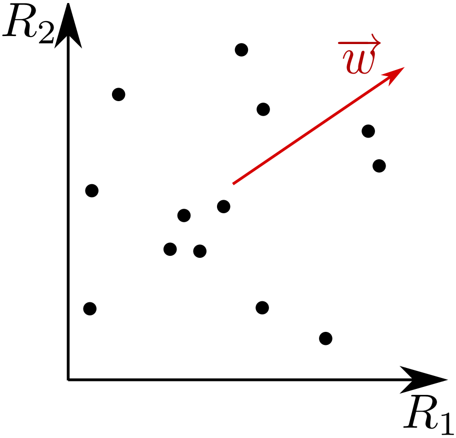
Q-Values in Space!

- $V(s_0)$
- $Q(s_0, a_0)$
- Each policy gives one value



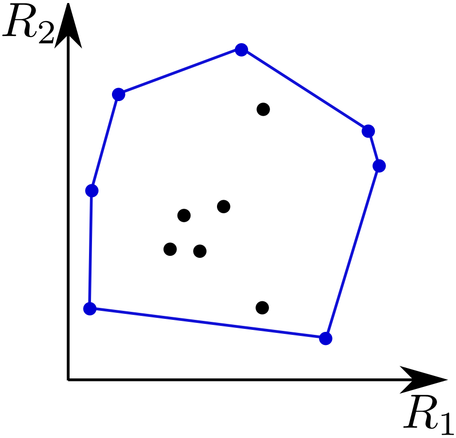
Q-Values in Space!

- $V(s_0)$
- $Q(s_0, a_0)$
- Each policy gives one value

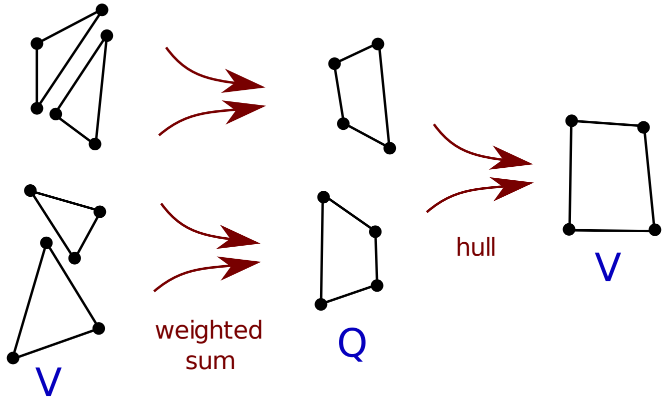


Q-Values in Space!

- $V(s_0)$
- $Q(s_0, a_0)$
- Each policy gives one value

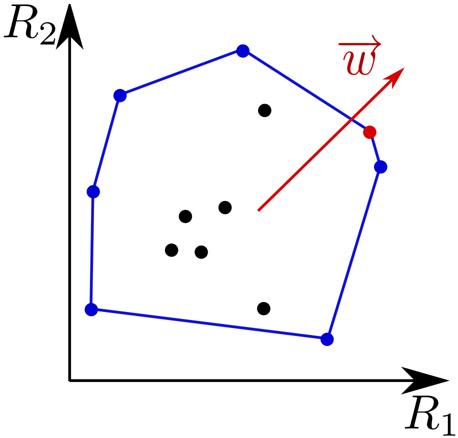


Revised Recurrences



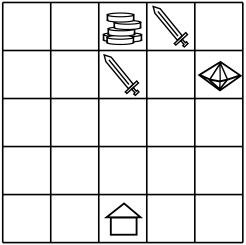
- $\overset{\circ}{Q}^{*}(s,a) = \mathbb{E}\left[\overrightarrow{R}(s,a) + \gamma \overset{\circ}{V}^{*}(s')\right]$
- $\overset{\circ}{V}^{*}(s) = \text{hull} \bigcup_a \overset{\circ}{Q}^{*}(s,a)$

Given a $\overrightarrow{w}$

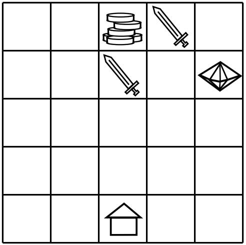


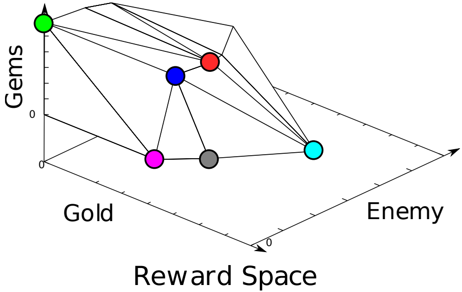
- Extract optimal value by taking max
- For all $\overrightarrow{w}$, solution identical to standard RL
  - Because max in any direction must be on hull

Example Results

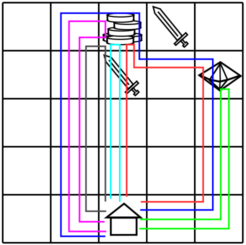


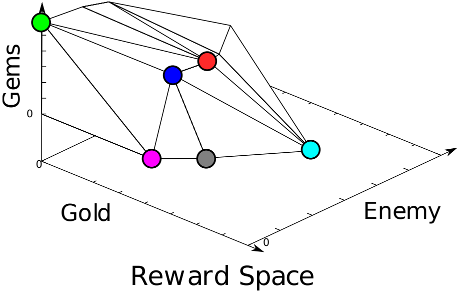
Example Results



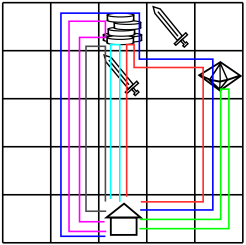


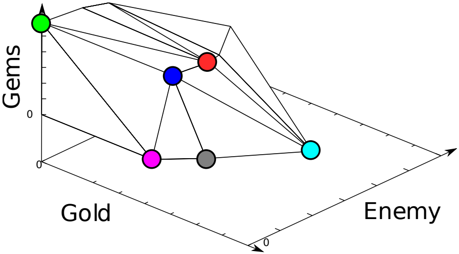
Example Results



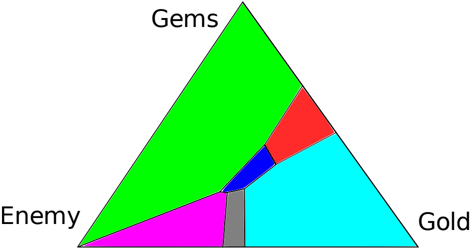


Example Results





Reward Space



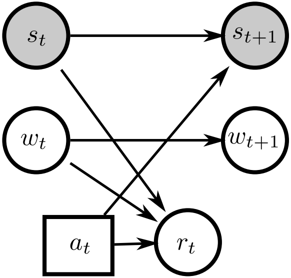
Preference Space

Complexity

- $O(n^d)$ for high dimension
- Efficient for 2D and 3D

- Efficiency tricks
  - Witnesses: check with previous hull
  - Constrain $\overrightarrow{w}$ space

POMDPs



- Rewrite as POMDP
- $\mathrm{P}(w) \leftrightarrow \overrightarrow{w}$

Contributions

- New class of results: *all* optimal policies
  - Via convex hull version of Bellman recurrence
  - Complete view of useful policy space
  - On-line preference switching

Future Work

- Combine with POMDPs
- Inverse problem: determine range of $\overrightarrow{w}$
  - Extract agent preferences
- Different discounting rates $\overrightarrow{\gamma}$
  - Approximate hyperbolic discounting

Thanks To

- My guinea pigs

Louis Milo Chester