Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning

Richard S. Sutton, Doina Precup, Satinder Singh

Presenter: Arseny Moskvichev

Overview

1) Motivation
2) MDPs, Options, and SMDPs
3) Results inherited from SMDPs
4) Option-specific results

Motivation for temporal abstraction

Motivation for temporal abstraction (airport example)

Motivation for temporal abstraction (airport example)

Motivation for temporal abstraction (airport example)

Motivation for temporal abstraction (airport example)

✓

What next?

Motivation for temporal abstraction (airport example)

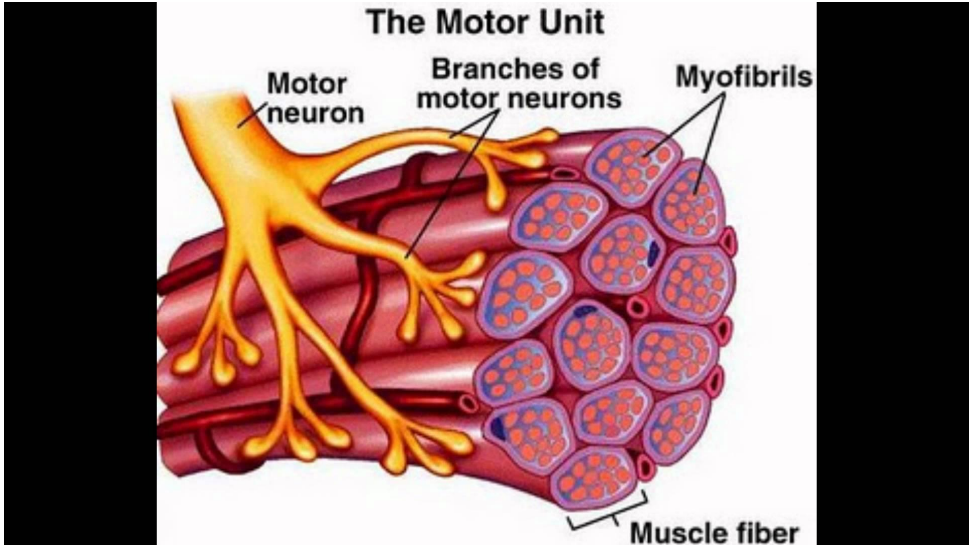
Make a call.

Motivation for temporal abstraction (airport example)

Which implies getting up and finding your phone

Motivation for temporal abstraction (airport example)

… Which implies contracting your muscles in a specific sequence



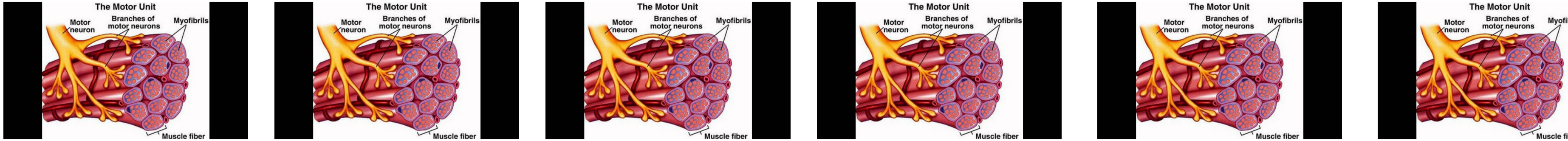
Motivation for temporal abstraction (airport example)

… etc.

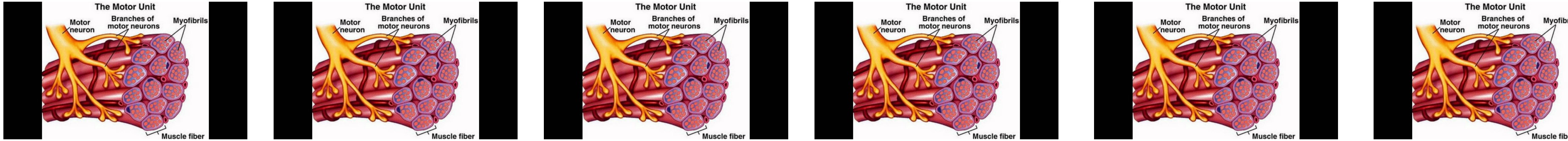
Now imagine for a second, how a trip to an airport would look like if we had no temporal abstraction mechanisms.

Motivation for temporal abstraction (airport example)

Should I walk?



Or take a taxi?



MDPs, SMDPs, and Options

Theoretical basis for temporal abstraction

MDPs

An MDP consists of:

A set of states

A set of actions,

Transition dynamics

$$p_{ss'}^{a} = \Pr\left\{ s_{t+1} = s' \,\middle|\, s_t = s,\ a_t = a \right\}$$

One-step expected reward

$$r_{s}^{a} = E\left\{ r_{t+1} \,\middle|\, s_t = s,\ a_t = a \right\}$$

MDPs

$$\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$$

$$\begin{aligned} V^{\pi}(s) &= E\left\{ r_{t+1} + \gamma r_{t+2} + \gamma^{2} r_{t+3} + \cdots \,\middle|\, s_t = s, \pi \right\} \\ &= E\left\{ r_{t+1} + \gamma V^{\pi}(s_{t+1}) \,\middle|\, s_t = s, \pi \right\} \\ &= \sum_{a \in \mathcal{A}_s} \pi(s, a) \left[r_s^a + \gamma \sum_{s'} p_{ss'}^{a} V^{\pi}(s') \right], \end{aligned}$$

MDPs

$$
\begin{aligned}
V^*(s) &= \max_{\pi} V^{\pi}(s) \\
&= \max_{a \in \mathcal{A}_s} E\Big\{ r_{t+1} + \gamma V^*(s_{t+1}) \,\Big|\, s_t = s,\ a_t = a \Big\} \\
&= \max_{a \in \mathcal{A}_s} \left[r_s^a + \gamma \sum_{s'} p_{ss'}^a V^*(s') \right].
\end{aligned}
$$

Semi MDP

Options

An option $\langle \mathcal{I}, \pi, \beta \rangle$ consists of three components:

A policy: $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$

A termination condition: $\beta : \mathcal{S}^{+} \rightarrow [0, 1]$

An initiation set: $\mathcal{I} \subseteq \mathcal{S}$

We say that the option is **available** at some state, if this state belongs to the option's initiation set.

Semi-markov Options

Like options, but decisions could depend on history since initiation,

$s_t, a_t, r_{t+1}, s_{t+1}, a_{t+1}, \ldots, r_\tau, s_\tau$, denoted $h_{t\tau}$

Therefore, now we need to re-define policy and termination conditions:

$\pi : \Omega \times \mathcal{A} \rightarrow [0, 1]$ and $\beta : \Omega \rightarrow [0, 1]$

Semi-markov Options

Semi markov options naturally arise when:

Semi-markov Options

Semi markov options naturally arise when:

- An option needs to terminate after being executed for some time, regardless of the resulting state
- Options have access to a more fine-grained representation of the state that the higher-level policy

Options. More definitions and details

$\mathcal{O}_s$ - like action sets, but for options

Viewing simple actions as single-step options

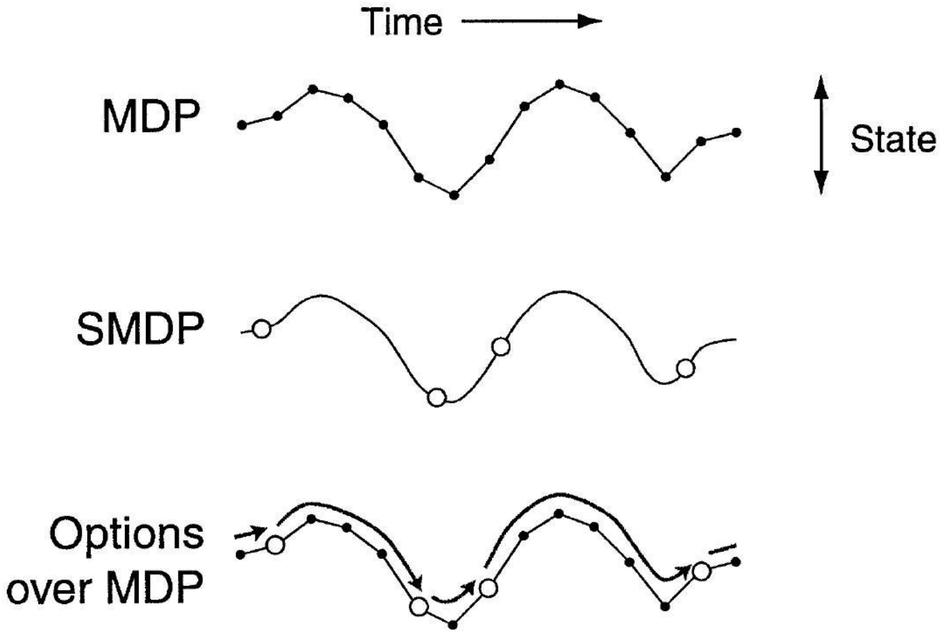
Composing options (note: composition of Markov options is semi-Markov)

Policies over options: $\mu : \mathcal{S} \times \mathcal{O} \to [0, 1]$

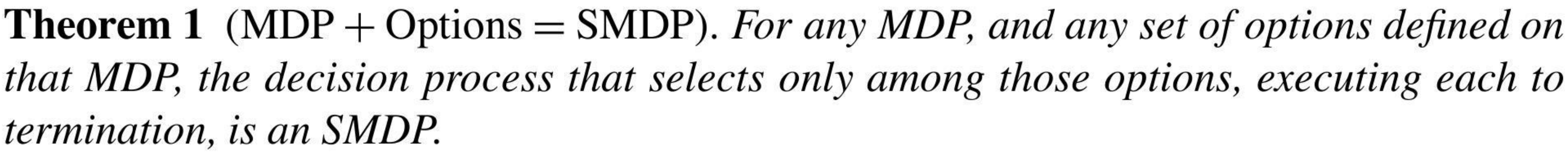
Associated flat policy: $\pi = flat(\mu)$

Note: even a policy composed of markov options is likely to be only semi-Markov.

MDPs, SMDPs, and Options (summary)



Relationship between Options and SMDPs



Theorem 1 (MDP + Options = SMDP). *For any MDP, and any set of options defined on that MDP, the decision process that selects only among those options, executing each to termination, is an SMDP.*

Rewriting Bellman equations with options

A few more definitions:

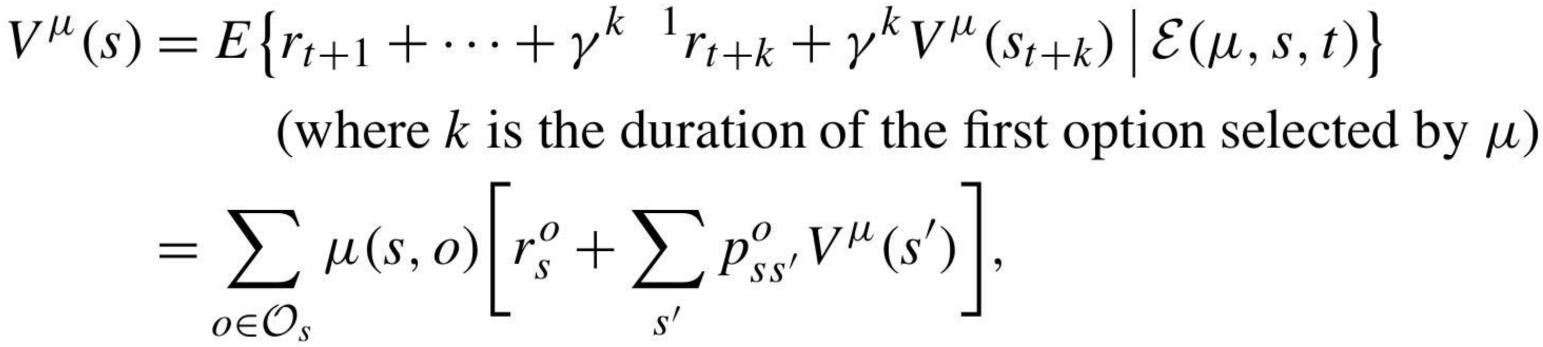
$$r_s^o = E\{r_{t+1} + \gamma r_{t+2} + \cdots + \gamma^{k\quad 1} r_{t+k} \,|\, \mathcal{E}(o, s, t)\}$$

$$p_{ss'}^o = \sum_{k=1}^{\infty} p(s', k)\gamma^k,$$

Where $p(s', k)$ is the probability that option terminates in a state s’ after k steps.

Rewriting Bellman equations with options

Using these definitions, we can finally write



$$V^{\mu}(s) = E\big\{r_{t+1} + \cdots + \gamma^{k \;\; 1} r_{t+k} + \gamma^{k} V^{\mu}(s_{t+k}) \,\big|\, \mathcal{E}(\mu, s, t)\big\}$$

(where k is the duration of the first option selected by μ)

$$= \sum_{o \in \mathcal{O}_s} \mu(s, o) \left[r_s^o + \sum_{s'} p_{ss'}^o V^{\mu}(s') \right],$$

Rewriting Bellman equations with options

$$
\begin{aligned}
Q^{\mu}(s,o) &= E\left\{r_{t+1} + \cdots + \gamma^{k\;\;1} r_{t+k} + \gamma^{k} V^{\mu}(s_{t+k}) \,\middle|\, \mathcal{E}(o,s,t)\right\}, \\
&= E\Bigg\{r_{t+1} + \cdots + \gamma^{k\;\;1} r_{t+k} \\
&\qquad + \gamma^{k} \sum_{o' \in \mathcal{O}_s} \mu(s_{t+k}, o') Q^{\mu}(s_{t+k}, o') \,\Bigg|\, \mathcal{E}(o,s,t)\Bigg\} \\
&= r_s^o + \sum_{s'} p_{ss'}^{o} \sum_{o' \in \mathcal{O}_{s'}} \mu(s', o') Q^{\mu}(s', o').
\end{aligned}
$$

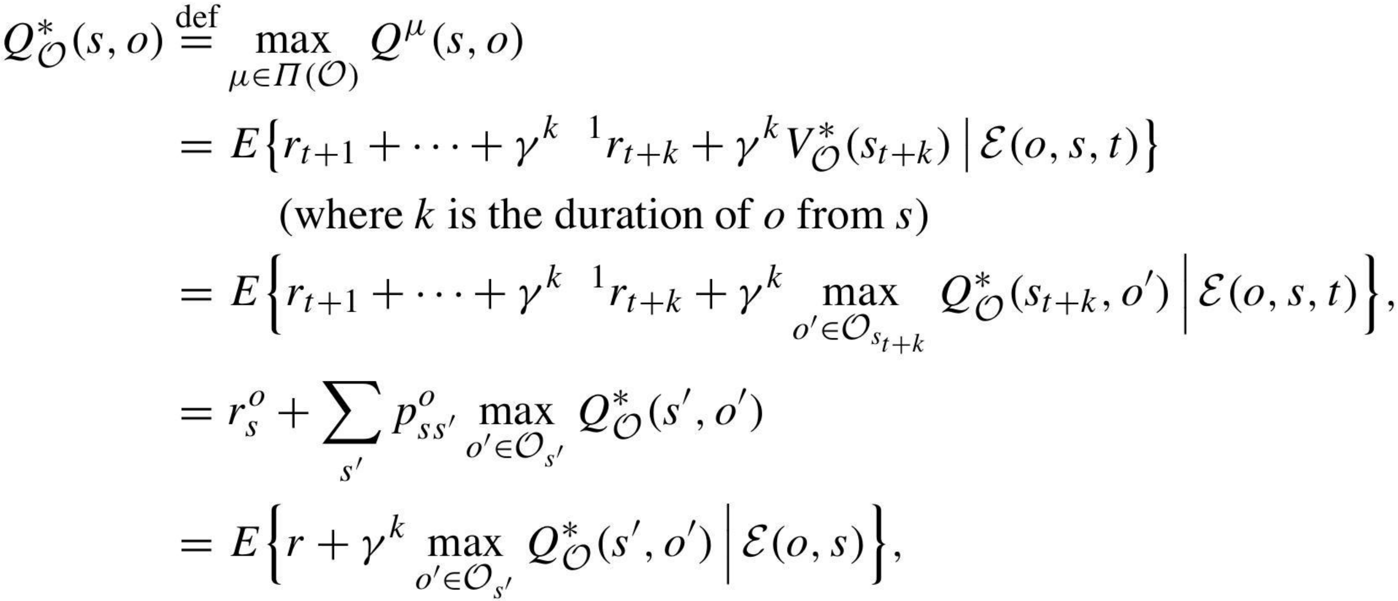
Rewriting Bellman equations with options

Similarly we can rewrite the bellman optimality equations.

$$
\begin{aligned}
V^*_{\mathcal{O}}(s) &\overset{\text{def}}{=} \max_{\mu \in \Pi(\mathcal{O})} V^{\mu}(s) \\
&= \max_{o \in \mathcal{O}_s} E\left\{ r_{t+1} + \cdots + \gamma^{k \quad 1} r_{t+k} + \gamma^k V^*_{\mathcal{O}}(s_{t+k}) \,\middle|\, \mathcal{E}(o, s, t) \right\} \\
&\qquad \text{(where } k \text{ is the duration of } o \text{ when taken in } s\text{)} \\
&= \max_{o \in \mathcal{O}_s} \left[r^o_s + \sum_{s'} p^o_{ss'} V^*_{\mathcal{O}}(s') \right] \\
&= \max_{o \in \mathcal{O}_s} E\left\{ r + \gamma^k V^*_{\mathcal{O}}(s') \,\middle|\, \mathcal{E}(o, s) \right\},
\end{aligned}
$$

Rewriting Bellman equations with options

Similarly we can rewrite the bellman optimality equations.

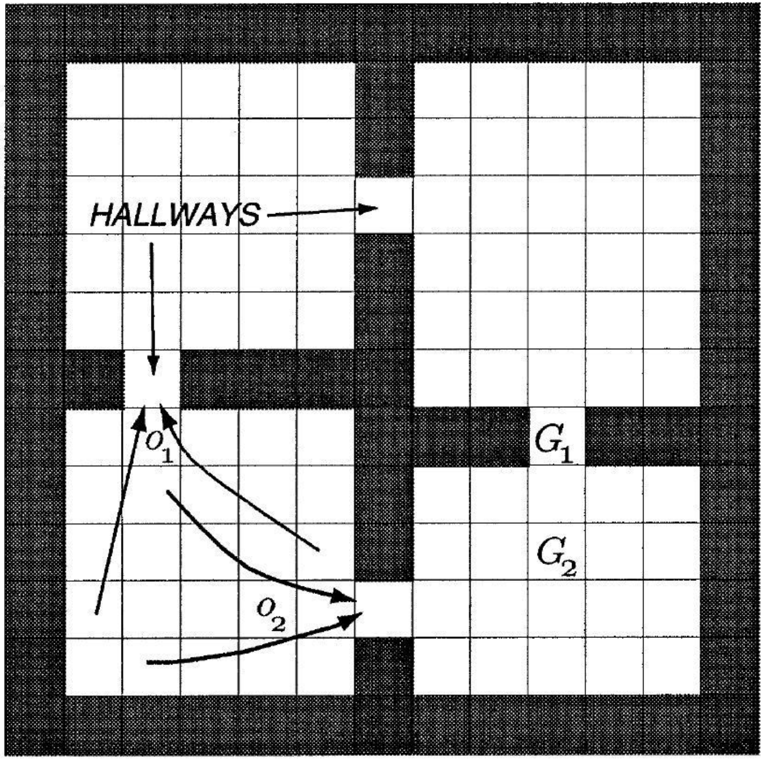


$$
\begin{aligned}
Q^*_{\mathcal{O}}(s,o) &\overset{\text{def}}{=} \max_{\mu\in\Pi(\mathcal{O})} Q^{\mu}(s,o) \\
&= E\left\{r_{t+1} + \cdots + \gamma^{k\quad 1} r_{t+k} + \gamma^{k} V^*_{\mathcal{O}}(s_{t+k}) \,\middle|\, \mathcal{E}(o,s,t)\right\} \\
&\qquad \text{(where } k \text{ is the duration of } o \text{ from } s\text{)} \\
&= E\left\{r_{t+1} + \cdots + \gamma^{k\quad 1} r_{t+k} + \gamma^{k} \max_{o'\in\mathcal{O}_{s_{t+k}}} Q^*_{\mathcal{O}}(s_{t+k},o') \,\middle|\, \mathcal{E}(o,s,t)\right\}, \\
&= r^o_s + \sum_{s'} p^o_{ss'} \max_{o'\in\mathcal{O}_{s'}} Q^*_{\mathcal{O}}(s',o') \\
&= E\left\{r + \gamma^{k} \max_{o'\in\mathcal{O}_{s'}} Q^*_{\mathcal{O}}(s',o') \,\middle|\, \mathcal{E}(o,s)\right\},
\end{aligned}
$$

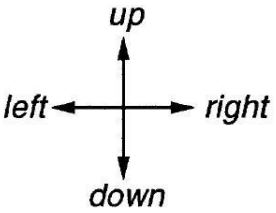
Why is it all useful?

Because we can now apply conventional DP algorithms to find policies over options.

Example



4 stochastic primitive actions

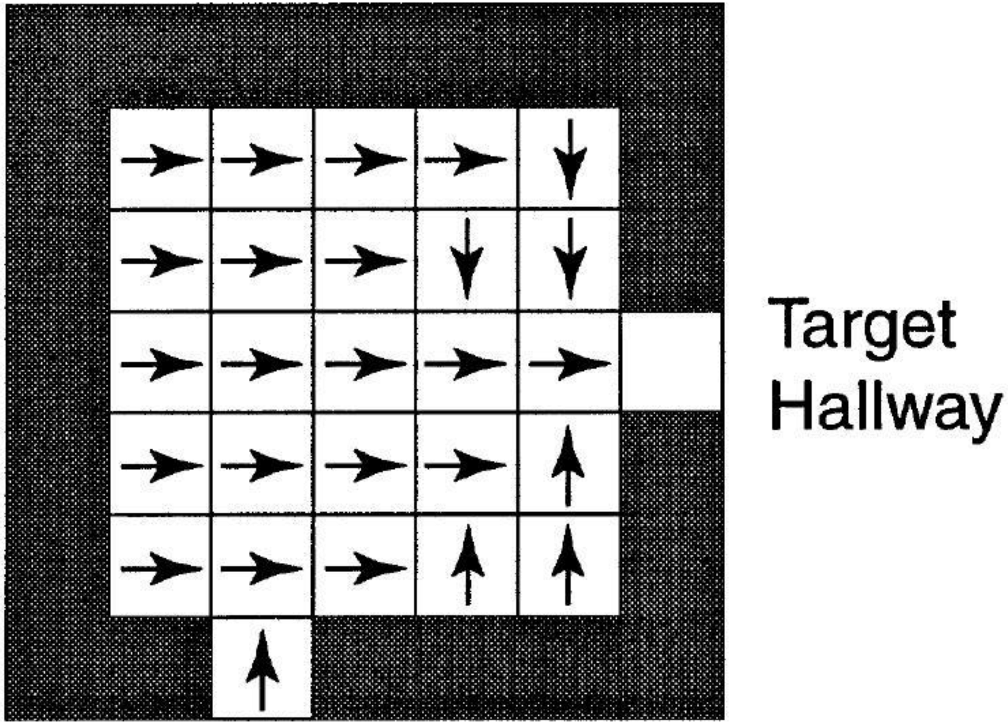


Fail 33% of the time

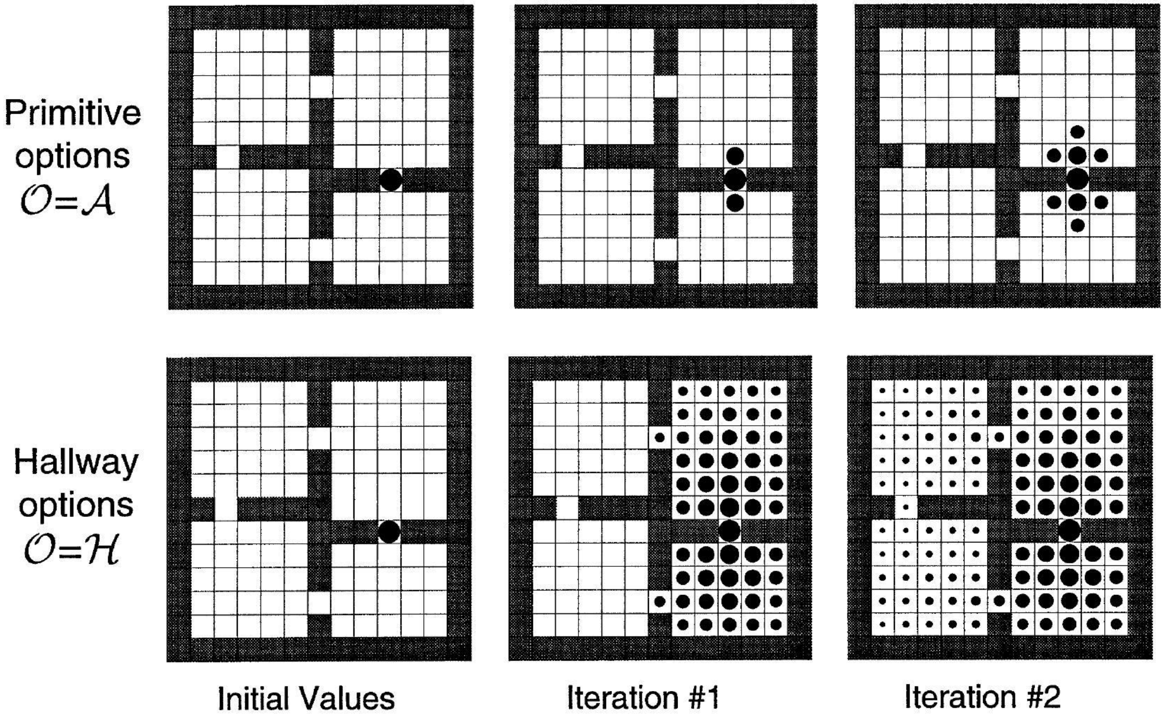
8 multi-step options

(to each room's 2 hallways)

Example (option)

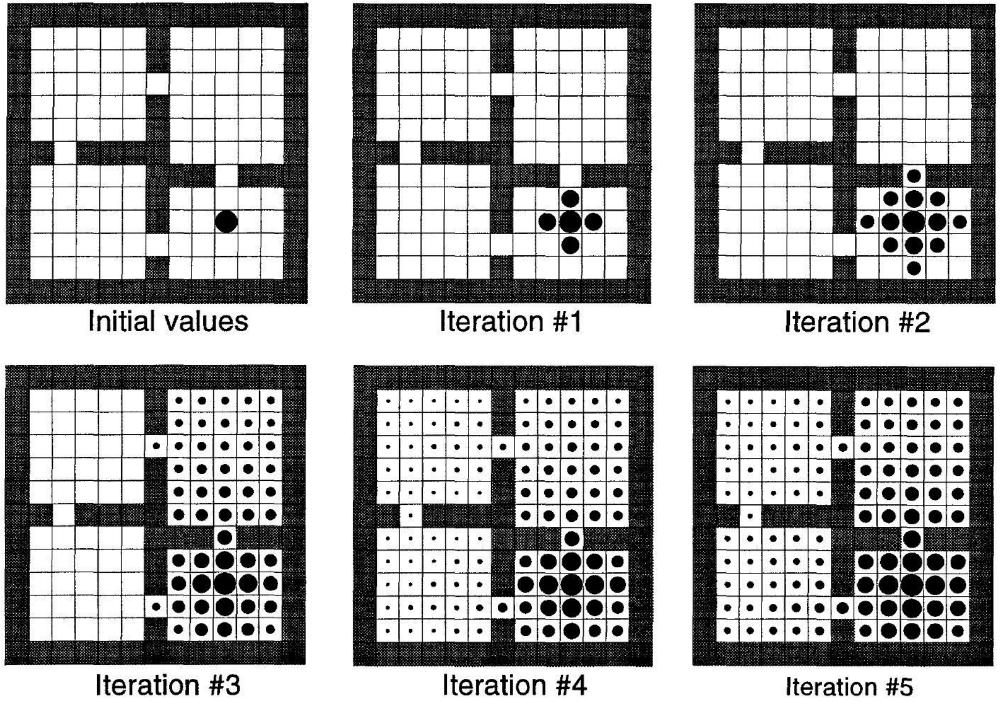


Example (hallway goal)

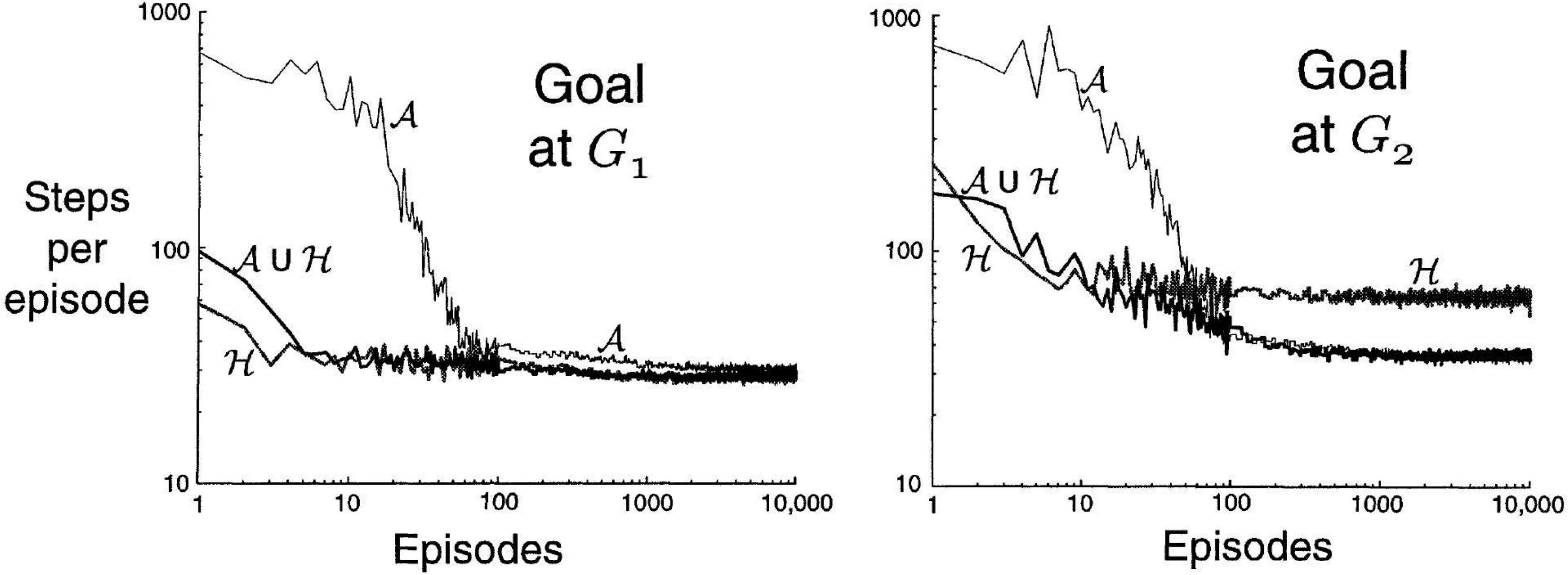


Example (non-hallway goal)

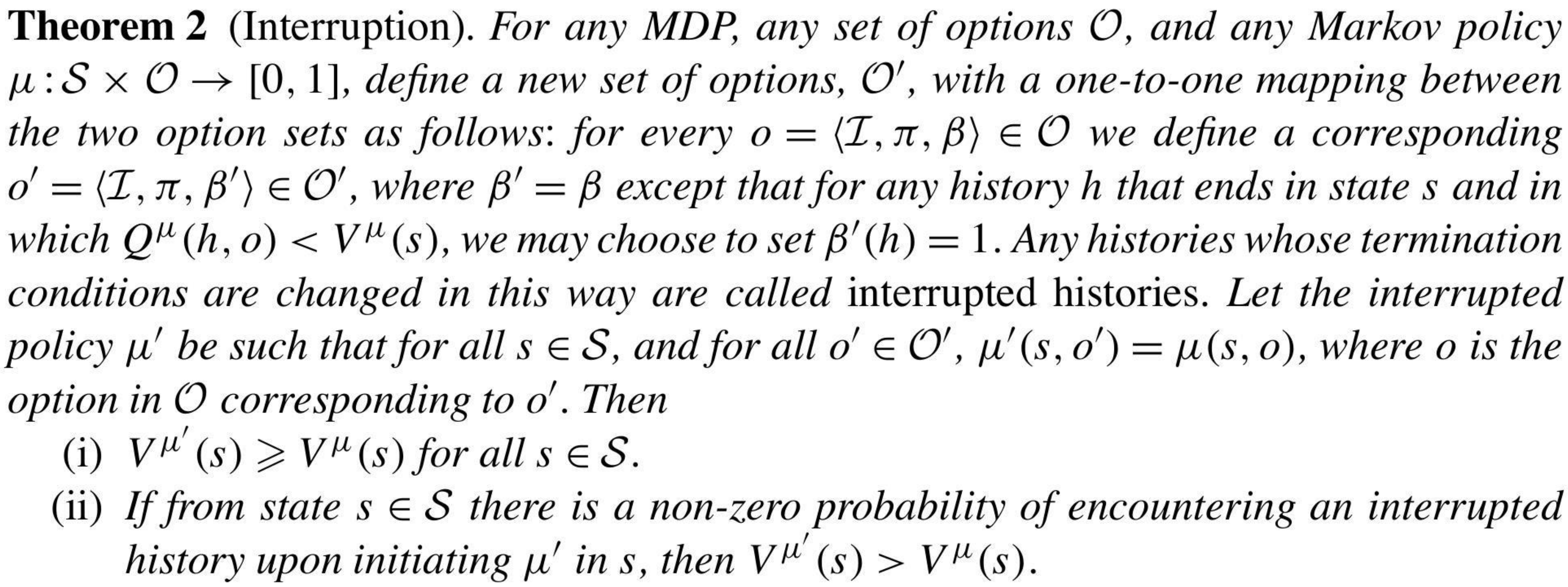
Primitive and hallway options $\mathcal{O}=\mathcal{A}\cup\mathcal{H}$



Learning


1000
100
10
Steps per episode
$\mathcal{A}$
$\mathcal{A} \cup \mathcal{H}$
$\mathcal{H}$
$\mathcal{A}$
Goal at G_1
1
10
100
1000
10,000
Episodes
1000
100
10
$\mathcal{A}$
$\mathcal{A} \cup \mathcal{H}$
$\mathcal{H}$
$\mathcal{H}$
Goal at G_2
1
10
100
1000
10,000
Episodes

Option-specific methods: Interruption



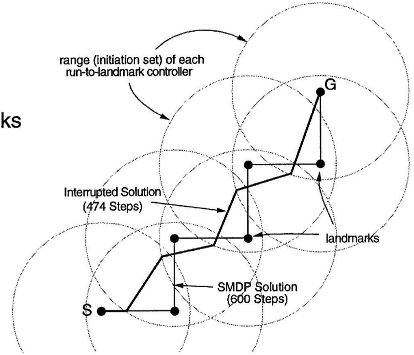
Theorem 2 (Interruption). *For any MDP, any set of options $\mathcal{O}$, and any Markov policy $\mu : \mathcal{S} \times \mathcal{O} \rightarrow [0, 1]$, define a new set of options, $\mathcal{O}'$, with a one-to-one mapping between the two option sets as follows: for every $o = \langle \mathcal{I}, \pi, \beta \rangle \in \mathcal{O}$ we define a corresponding $o' = \langle \mathcal{I}, \pi, \beta' \rangle \in \mathcal{O}'$, where $\beta' = \beta$ except that for any history h that ends in state s and in which $Q^{\mu}(h, o) < V^{\mu}(s)$, we may choose to set $\beta'(h) = 1$. Any histories whose termination conditions are changed in this way are called* interrupted histories. *Let the interrupted policy μ' be such that for all $s \in \mathcal{S}$, and for all $o' \in \mathcal{O}'$, $\mu'(s, o') = \mu(s, o)$, where o is the option in $\mathcal{O}$ corresponding to o'. Then*

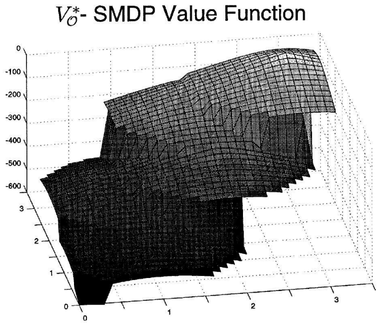
(i) $V^{\mu'}(s) \geqslant V^{\mu}(s)$ *for all* $s \in \mathcal{S}$.

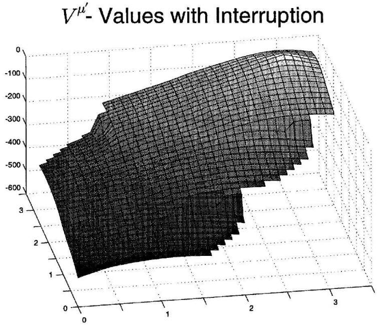
(ii) *If from state $s \in \mathcal{S}$ there is a non-zero probability of encountering an interrupted history upon initiating μ' in s, then $V^{\mu'}(s) > V^{\mu}(s)$.*

Option-specific methods: Interruption

Landmarks
Problem



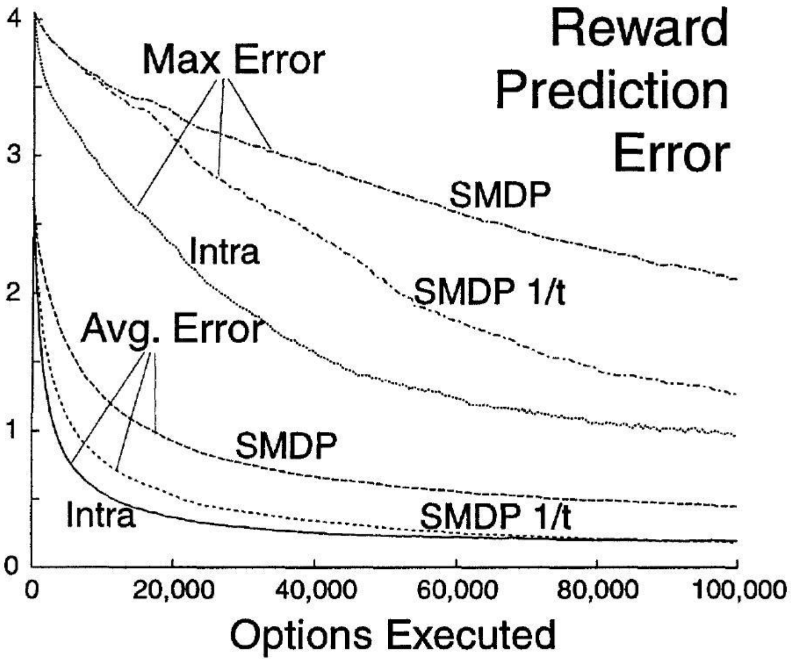

$V^*_{\mathcal{O}}$- SMDP Value Function

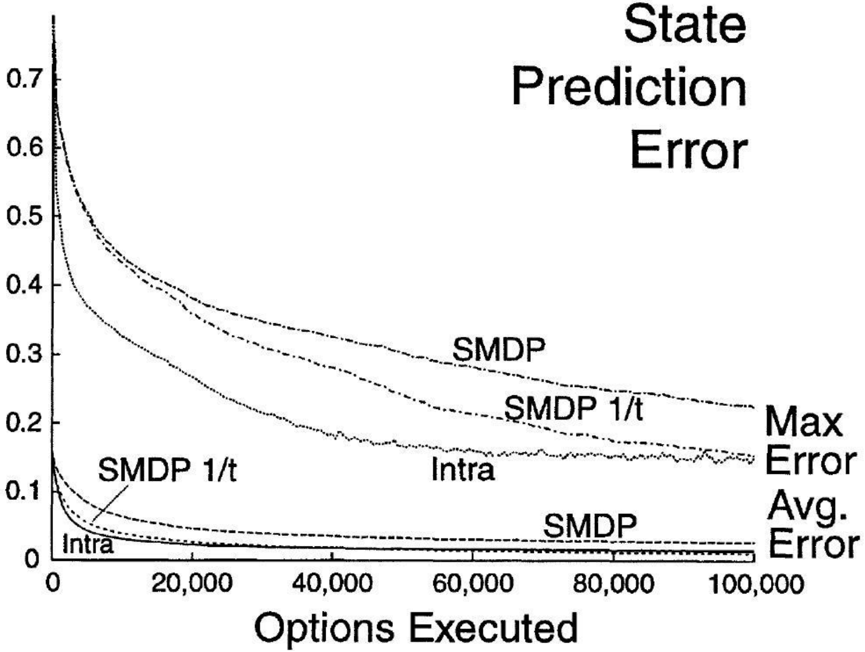

$V^{\mu'}$- Values with Interruption

Option-specific methods:

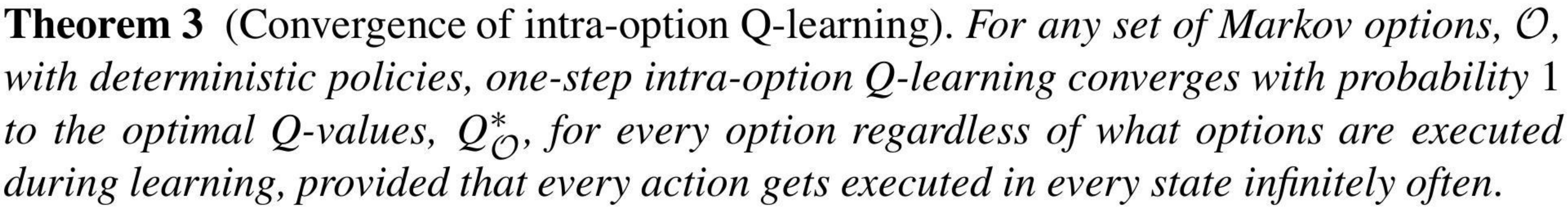
Intra-option **model** learning and
Intra-option **value** learning
Subgoals for learning options

Intra-option model learning:





Intra-option value learning:



Theorem 3 (Convergence of intra-option Q-learning). *For any set of Markov options, $\mathcal{O}$, with deterministic policies, one-step intra-option Q-learning converges with probability* 1 *to the optimal Q-values, $Q^*_{\mathcal{O}}$, for every option regardless of what options are executed during learning, provided that every action gets executed in every state infinitely often.*

Subgoals for learning options:

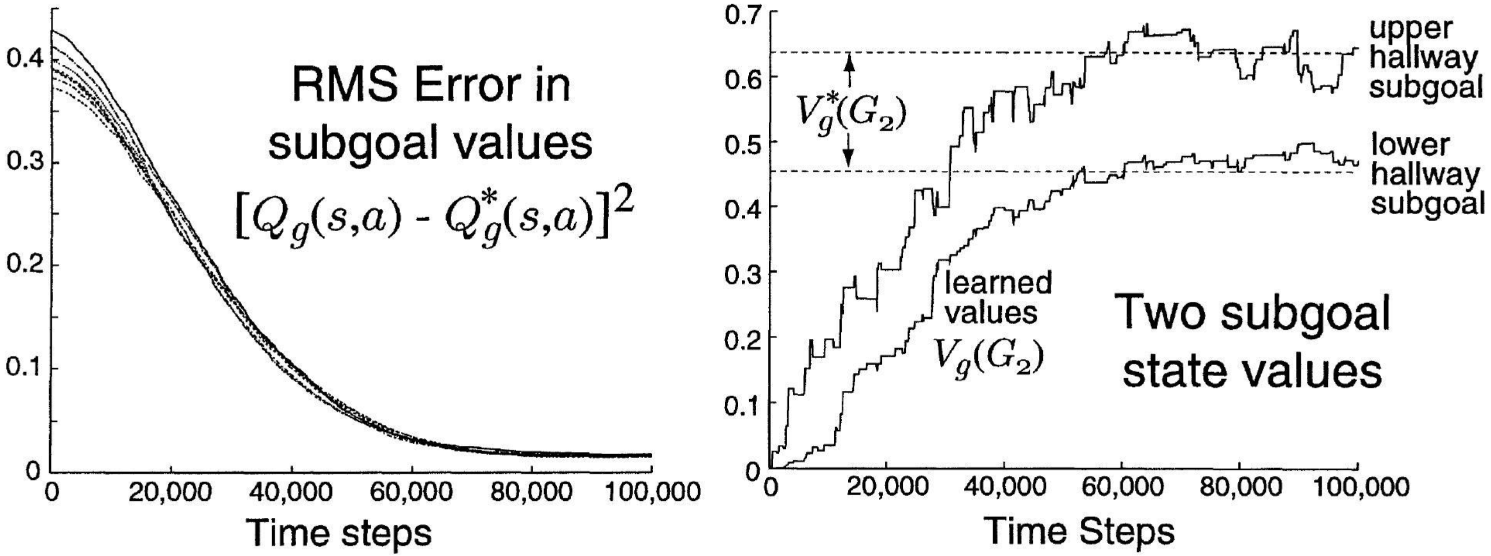


Fig. 11. Learning subgoal-achieving hallway options under random behavior. Shown on the left is the error between $Q_g(s,a)$ and $Q^*_g(s,a)$ averaged over $s \in \mathcal{I}$, $a \in \mathcal{A}$, and 30 repetitions. The right panel shows the learned state values (maximum over action values) for two options at state G_2 approaching their correct values.

Summary

Options are temporally extended actions with initiation sets and termination conditions. Options defined over an MDP yield SMDPs and all SMDP methods and results apply.

Options compare favorably to SMDPs, since they also allow to leverage our knowledge about the structure of a temporally-extended actions. In particular, option-specific methods include:

- Interrupting options
- Intra-option model/value learning
- Learning with subgoals