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

Paper by: R.Sutton, Doina Precup, Satinder Singh

Monica Patel (260728093)

31-March-2017

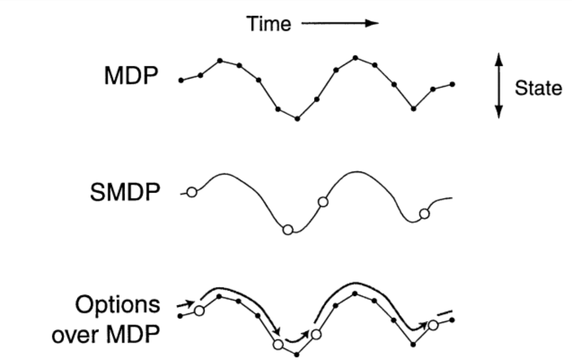
McGill University

Synopsis of the Paper

- Paper introduces extended notion of actions from MDP frame work to include 'options', to take action over a period of time.
- Paper also shows that options can be used like actions for planning and learning.
- There is brief introduction to what are options and what are semi-Markov decision process.
- Notion of subgoal is introduced which can be used to improve options.

MDP and SMDP frame work

- The base problem under consideration is a MDP, but the state transition resulted from actions of this MDP are extended and variable in time. These actions are called OPTIONS.
- Fixed Set of such options defines a discrete time SMDP, which is embedded within MDP.



Options

- Options are generalization of temporally extended actions in MDP.
- It consists of three components:
  - Policy π: S x A $\rightarrow$ [0,1]
  - Termination condition β: $S^+ \rightarrow$ [0,1]
  - initiation set: I $\subset$ S
- An option is available in state s_t iff $s_t \in I$
- If the option is taken, then actions are selected according to π until the option terminates stochastically according to β
- Its is also helpful sometimes to terminate or exit an option after a particular time τ even if it fails to reach the end states.
- Since the above is not possible in Markov option, (termination depends solely on current state) Semi- Markov options are used.
- in Semi-Markov options Policy and termination condition is decided based on history for the option was running: $h_{t\tau}$ (running from t to τ)
- Set of all history is denoted by Ω

Options Continued

- Given two options they can be taken in sequence. Taking b after a terminates (inside of b's initiation states)
- This is a composed option ab.
- Composition of two Markov options is semi-Markov option because actions are selected differently before and after termination of first option.
- Composition of two semi-Markov options is semi-Markov option.

Policies over Options

Markov Policy over options: S x O $\rightarrow$ [0,1]

- Initialize in state s_t
- Select an option o $\in$ O according to probability distribution $\mu(s_t, .)$
- Option o is taken in s_t until it terminates in state s_{t+k} at which time new option is selected and this goes on.
- Value of state s under semi Markov flat policy is defined as expected return given that π initialized in s.
  $V^\pi = E\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} ... | e(\pi, s, t)\}$
  where $e(\pi, s, t)$ denotes event of π initiated in s at time t.
- Option-Value function $Q^\mu(s, o)$ is defined as the value of taking option o in state s $\in$ I under policy π
  $Q^\mu(s, o) = E\{r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} ... | e(o\mu, s, t)\}$
  Where $o\mu$ is composition of o and μ, first follow o then choose next option according to μ

V* and Q* over Options

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

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

SMDP Planning

The above bellman equations can used for dynamic-programming style planning for SMDP. Typically approximation of V* and Q* is maintained for all states and options.
Synchronous Value Iteration:

- Option starts with arbitrary approximation of V* and computes new approximation by:
  $V_k(s) = max_{o \in O_s}[r_s^o + \sum_{s' \in S} P_{ss'}^o V_k 1(s')]$
- New approximation for Q* are computed by
  $Q_k(s, o) = r_s^o + \sum_{s' \in S} P_{ss'}^o max_{o' \in O_s} Q_k 1(s', o')$

Conclusion

- Representing knowledge at multiple level of temporal abstraction speed up planning and learning.
- Transfer between subtask is not completely understood.
- As for extended actions there can be same implications made for extended perception.