Hierarchical Policy Gradient Algorithms

Hierarchical Policy Gradient Algorithms (http://blogs.cuit.columbia.edu/zp2130/files/2019/02/Hierarchical-Policy-Gradient-Algorithms.pdf)

Math

Notation

M : the overall task MDP.

$\{M^0, M^1, M^2, M^3, \ldots, M^n\}$: a finite set of subtask MDPs.

M^i : subtask, models a subtask in the hierarchy.

M^0 : root task and solving it solves the entire MDP M.

i : *non-primitive* **subtask**, paper uses ***subtask*** to refer to *non-primitive* **subtask**.

$\boldsymbol{S}^i$: state space for non-primitive subtask i.

I^i : initiation set.

T^i : set of terminal state.

A^i : action space.

P^i : transition probability function.

R^i : reward function.

a : **primitive** action, is a primitive subtask in this decomposition, it terminates immediately after execution.

μ^i : subtask *i* **policy.**

$\mu=\{\mu^0, \mu^1, \mu^2, \mu^3, \ldots, \mu^n\}$: ***hierarchical policy***

$\mu^i(\theta^i)$: randomized **stationary policies** parameterized in terms of a vector $\theta^i \in \Re^K$.

$\mu^i(s, a, \theta^i)$: **probability** of taking action a in **state s** under the **policy** corresponding to θ^i.

s^{*i} : absorbing **state**, all terminal states ($s \in T^i$) transit with probability 1 and reward 0 to an absorbing state s^{*i} .

$\bar{\pi}^i(s)$: The probability that subtask *i* **starts** at **state *s***.

$M^i_{\bar{\pi}^i}$: MDP for ***subtask i*** .

$P^i{}_{\bar{\pi}^i}(s' \mid s, a) = \begin{cases} P^i(s' \mid s, a) & s \neq s^{*i} \\ \bar{\pi}^i(s') & s = s^{*i} \end{cases}$: MDP for subtask *i* **transition probabilities**.

$\mathbb{P}^i_{\bar{\pi}}$: the set of all **transition** matrices $P^i{}_{\bar{\pi}^i}(s' \mid s, \mu^i(\theta^i))$.

$\chi^i(\theta^i)$: **weighted reward-to-go**, the performance measure of ***subtask i*** formulated by **parameterized policy $\mu^i(\theta^i)$.**

$J^i(s, \theta^i)$: **reward-to-go** of **state s**.

$T = min\{k > 0 \mid s_k = s^{*i}\}$: the first future time that state s^{*i} is visited.

$\nabla\chi^i(\theta^i)$: gradient of the *weighted* ***reward-to-go*** $\chi^i(\theta^i)$ with respect to θ^i.

$\pi^i_{\bar{\pi}}(s, \theta^i)$: steady state **probability distribution** of being in state s.

$E_{\bar{\pi},\theta^i}[T] = E_{\bar{\pi}^i,\theta^i}[\, T \mid s_0 = s^{*i} \,]$: mean recurrence time.

$Q^i(s, a, \theta^i) = E_{\bar{\pi}^i,\theta^i}\left[\sum_{k=0}^{T-1} R^i_{\bar{\pi}^i}(s_k, \theta^i) \mid s_0 = s, a_0 = a\right]$: usual **action-value function.**

$F^i_m(\theta^i)$: **estimate** gradient of the *weighted* ***reward-to-go*** $\chi^i(\theta^i)$ with respect to θ^i , i.e., $\nabla\chi^i(\theta^i)$.

t_m : the time of the *m*th visit at the recurrent state s^{*i} .

$\tilde{Q}^i(s_n, a_n, \theta^i) = \sum_{k=n}^{t_{m+1}-1} R^i(s_n, a_n)$: an estimate of Q^i .

α : step size parameter.

Policy Gradient Formulation

After decomposing the overall problem into a set of subtasks, the paper formulates each subtask as policy gradient reinforcement learning problem. The paper focus on ***episodic*** problems, so it assume that the **overall task** (root of the hierarchy) is ***episodic***.

Assumption 1

For every state $s \in S^i$ and every action $a \in A^i$, $\mu^i(s, a, \theta^i)$ as a function of θ^i , is bounded and has bounded first and second derivatives. Furthermore, we have

$$\nabla\mu^i(s, a, \theta^i) = \mu^i(s, a, \theta^i)\ \psi^i(s, a, \theta^i)$$

where $\psi^i(s, a, \theta^i)$ is bounded, differential and has bounded first derivatives.

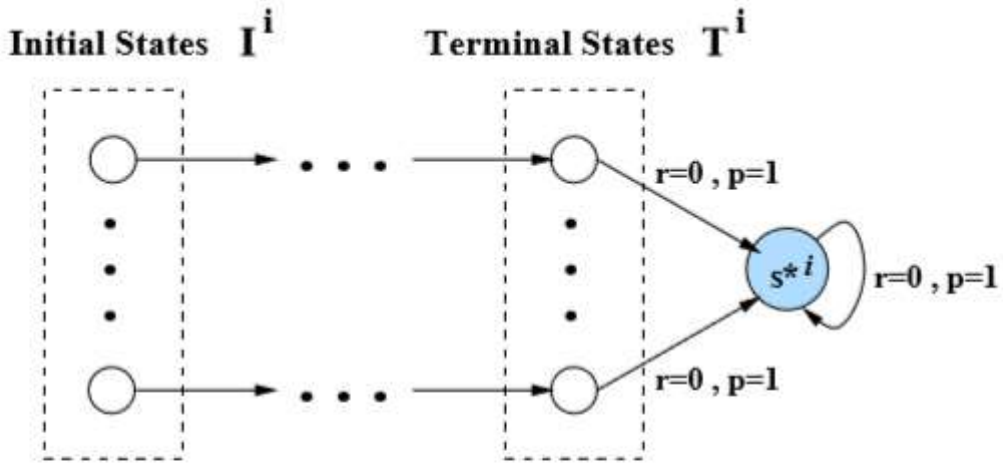

Figure 4. This figure shows how we model a subtask as an *episodic* problem under assumption A2.

Assumption 2

Subtask Termination

There exists a state $s^{*i} \in \boldsymbol{S^i}$ such that, for every action $a \in A^i$, we have

$$R^i(s^{*i}, a) = 0$$

$$P^i(s^{*i} \mid s^{*i}, a) = 1$$

and for all stationary polices $\boldsymbol{\mu^i(\theta^i)}$ and all states $s^i \in \boldsymbol{S^i}$, we have

$$P^i(s^{*i}, N \mid s, \mu^i(\theta^i)) > 0$$

$$N = \left|S^i\right|$$

Under this model, we define a new MDP $M^i_{\bar{\pi}^i}$ for ***subtask i* transition probabilities:**

$$P^i{}_{\bar{\pi}^i}(s' \mid s, a) = \begin{cases} P^i(s' \mid s, a) & s \neq s^{*i} \\ \bar{\pi}^i(s') & s = s^{*i} \end{cases}$$

where

$\bar{\pi}^i(s)$: the **probability** that ***subtask i*** starts at state *s.*

Let $\mathbb{P}^i_{\bar{\pi}}$ be the set of all transition matrices $P^i{}_{\bar{\pi}^i}(s' \mid s, \mu^i(\theta^i))$. We have the following result for ***subtask i***.

Lemma 1

Let assumptions A1 and A2 hold. Then for every $P^i_{\bar{\pi}^i} \in \mathbb{P}^i_{\bar{\pi}}$, and every state $s \in S^i$, we have

$\sum_{n=1}^{N} P^i_{\bar{\pi}}(s^{*i}, n \mid s, \mu^i(\theta^i)) > 0$where

$$N = \left|S^i\right|$$

Lemma 1 is equivalent to assume that the MDP $M^i_{\bar{\pi}^i}$ is **recurrent**, i.e., the underlying Markov chain for every **policy $\boldsymbol{\mu^i(\theta^i)}$** in this MDP has a **single recurrent class** and the state $\boldsymbol{s^{*i}}$ is recurrent state.

Performance Measure Definition

Paper defines **weighted reward-to-go** $\chi^i(\theta^i)$, the performance measure of ***subtask i*** formulated by **parameterized policy $\boldsymbol{\mu^i(\theta^i)}$.**

Assumption A2 holds

$$\chi^i(\theta^i) = \sum_{s \in S^i} \bar{\pi}^i(s) J^i(s, \theta^i)$$

where $J^i(s, \theta^i)$ is **reward-to-go** of **state s**.

$$J^i(s, \theta^i) = E_{\theta^i}\left[\sum_{k=0}^{T-1} R^i(s_k, \theta^i) \mid s_0 = s\right]$$

$$J^i(s, \theta^i) = E_{\theta^i}\left[\sum_{k=0}^{T-1} R^i(s_k, \theta^i) \mid s_0 = s\right]$$

where T is the first future time that state $\boldsymbol{s^{*i}}$ is visited.

$$T = min\left\{k > 0 \mid s_k = s^{*i}\right\}$$

Optimizing the Weighted Reward-to-Go

Proposition 1

If assumptions A1 and A2 hold

$$\bigtriangledown \chi^i(\theta^i) = E_{\bar{\pi}^i}[T] \sum_{s \in S^i} \sum_{a \in A^i} \pi^i_{\bar{\pi}^i}(s, \theta^i) \bigtriangledown \mu^i(s, a, \theta^i) Q^i(s, a, \theta^i)$$

```
Cycle between consecutive visits to recurrent state s*i = renewal cycle
```

$\bigtriangledown \chi^i(\theta^i)$ can be **estimated** over a **renewal cycle** as

$$F_m^i(\theta^i) = \sum_{n=t_m}^{t_{m+1}-1} \tilde{Q}^i(s_n, a_n, \theta^i) \frac{\nabla \mu^i(s_n, a_n, \theta^i)}{\mu^i(s_n, a_n, \theta^i)} \quad \textbf{(P1)}$$

$$\tilde{Q}^i(s_n, a_n, \theta^i) = \sum_{k=n}^{t_{m+1}-1} R^i(s_n, a_n)$$

where t_m is the time of the mth visit at the recurrent state s^{*i} .

From **P1**, the paper obtains the following procedure to update the parameter vector along the **approximate gradient direction** at every time step.

$$z^i{}_{k+1} = \begin{cases} 0 & s_k = s^{*i} \\ z_k^i + \psi^i(s_k, a_k, \theta_k^i) & otherwise \end{cases} \quad \textbf{(P2)}$$
$$\theta^i{}_{k+1} = \theta_k^i + \alpha_k^i R^i(s_k, a_k) z^i{}_{k+1}$$

P2 provides an unbiased estimate of $\nabla \chi^i(\theta^i)$. For systems involving a large state space, the **interval** between visits to state s^{*i} can be **large**. As a consequence, the estimate of $\nabla \chi^i(\theta^i)$ might have a **large variance**.

α : step size parameter and satisfies the following assumptions.

Assumption 4

α_k 's are deterministic, nonnegative and satisfy

$$\sum_{k=1}^{\infty} \alpha_k = \infty \; and \; \sum_{k=1}^{\infty} \alpha_k^2 < \infty$$

Assumption 5

α_k 's are non-increasing and there exists **a positive integer p** and **a positive scalar A** such that

$$\sum_{k=n}^{n+t} (\alpha_n - \alpha_k) \leq A t^p {\alpha_n}^2$$

for **all positive integers** n and t.

The paper has the following **convergence result** for the iterative procedure in **P2 to update the parameters.**

Proposition 2

Let assumption A1, A2, A4 and A5 hold, and let $\left(\theta_k^i\right)$ be the **sequence of parameter vectors** generated by **P2**. Then, $\chi^i(\theta_k^i)$ **converges** and

$$lim_{k \to \infty} \nabla \chi^i(\theta_k^i) = 0$$

with probability 1.

---

edit (https://blogs.cuit.columbia.edu/zp2130/wp-admin/post.php?post=4778&action=edit)

Author: Z Pei (https://blogs.cuit.columbia.edu/zp2130/author/zp2130/) on February 26, 2019

Categories: AI (https://blogs.cuit.columbia.edu/zp2130/category/ai/), Algorithm (https://blogs.cuit.columbia.edu/zp2130/category/algorithm/), Machine Learning (https://blogs.cuit.columbia.edu/zp2130/category/machine-learning/), Policy Gradient Methods (https://blogs.cuit.columbia.edu/zp2130/category/policy-gradient-methods/), Reinforcement Learning (https://blogs.cuit.columbia.edu/zp2130/category/reinforcement-learning/), RL (https://blogs.cuit.columbia.edu/zp2130/category/rl/)

Tags: AI (https://blogs.cuit.columbia.edu/zp2130/tag/ai/), Hierarchical RL (https://blogs.cuit.columbia.edu/zp2130/tag/hierarchical-rl/), Machine Learning (https://blogs.cuit.columbia.edu/zp2130/tag/machine-learning/), Policy Gradient Methods (https://blogs.cuit.columbia.edu/zp2130/tag/policy-gradient-methods/), Reinforcement Learning (https://blogs.cuit.columbia.edu/zp2130/tag/reinforcement-learning/), RL (https://blogs.cuit.columbia.edu/zp2130/tag/rl/)

Other posts

Hierarchical Actor-Critic (https://blogs.cuit.columbia.edu/zp2130/hierarchical_actor-critic/) «» Decentralized Stabilization for a Class of Continuous-Time Nonlinear Interconnected Systems Using Online Learning Optimal Control Approach (https://blogs.cuit.columbia.edu/zp2130/decentralized_stabilization_for_a_class_of_continuous-time_nonlinear_interconnected_systems_using_online_learning_optimal_control_approach/)

Last posts

- Symbolic Netlist to Innovus-friendly Netlist (https://blogs.cuit.columbia.edu/zp2130/symbolic_netlist_to_innovus-friendly_netlist/)
- Finite-Sample Convergence Rates for Q-Learning and Indirect Algorithms (https://blogs.cuit.columbia.edu/zp2130/finite-sample_convergence_rates_for_q-learning_and_indirect_algorithms/)
- Solving H-horizon, Stationary Markov Decision Problems In Time Proportional To Log(H) (https://blogs.cuit.columbia.edu/zp2130/paul_tseng_1990/)

- Randomized Linear Programming Solves the Discounted Markov Decision Problem In Nearly-Linear (Sometimes Sublinear) Run Time (https://blogs.cuit.columbia.edu/zp2130/randomized_linear_programming_solves_the_discounted_markov_decision_problem_in_nearly-linear_sometimes_sublinear_run_time/)
- KL Divergence (https://blogs.cuit.columbia.edu/zp2130/kl_divergence/)
- The Asymptotic Convergence-Rate of Q-learning (https://blogs.cuit.columbia.edu/zp2130/the_asymptotic_convergence-rate_of_q-learning/)
- Hierarchical Apprenticeship Learning, with Application to Quadruped Locomotion (https://blogs.cuit.columbia.edu/zp2130/hierarchical_apprenticeship_learning_with_application_to_quadruped_locomotion/)
- Policy Gradient Methods (https://blogs.cuit.columbia.edu/zp2130/policy_gradient_methods/)
- Actor-Critic Algorithms for Hierarchical Markov Decision Processes (https://blogs.cuit.columbia.edu/zp2130/actor-critic_algorithms_for_hierarchical_markov_decision_processes/)
- Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation (https://blogs.cuit.columbia.edu/zp2130/hierarchical_deep_reinforcement_learning_integrating_temporal_abstraction_and_intrinsic_motivation/)