Peter Stone

Monday, 22 February 2021 22:38

① Robot skill learning: from real-world to simulation and back.

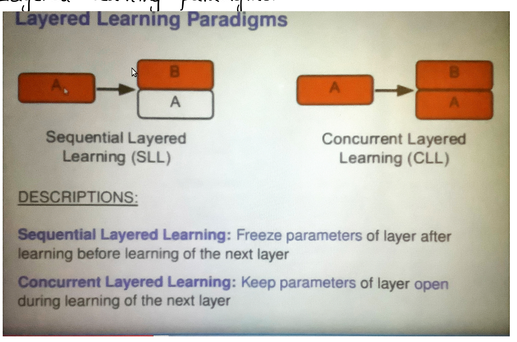
- Research question:

  To what degree can autonomous intelligent agents learn in the presence of teammates and/or adversaries in real-time dynamic domain?

- Layered learning:
  - For domains too complex for tractably mapping state features S to outputs O.
  - Hierarchical subtask decomposition given $\{L_1, \dots, L_n\}$
  - Machine learning: exploit data to train, adapt

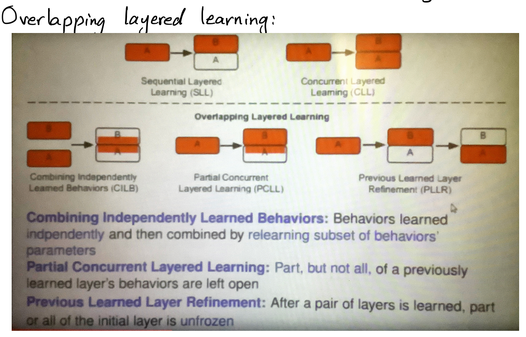
  Learning in one layer feeds into next layer.

- Layered learning paradigms:



Problems: SLL: can be too limiting in the joint layer policy search space.
CLL: The increased dimensionality can make learning harder.

- Overlapping layered learning:



- Grounded Simulation learning:

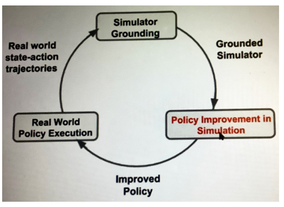
  modifying the simulator with real-world data so that policies learned in simulation work in the real-world.

  - idea: replace robot's action a_t with an action that produces a more realistic transition,

    → action as a function $g_\phi(s_t, a_t) = f^{-1}_{sim}(s_t, f(s_t, a_t))$

    simulator inverse dynamics — robot dynamics

  - Policy optimization with CMA-ES. learned using supervised learning



- TEXPLORE for Robot RL [Hester & Stone '13]
- Ad Hoc Teams: Ad hoc team player is an individual
  - Unknown team mates.
  - Teammates likely sub-optimal: no control.

  [Genter et al. '15]

② Efficient Robot skill learning:

- Off-policy policy evaluation:

  performance: $v(\pi) = \mathbb{E}[\sum_{t=1}^{L} R_t \mid H \sim \pi]$ (horizon)

  evaluation: given a target policy π, and data generated from a behavior policy π_b, estimate $v(\pi) \to \hat{v}$

  metric: $MSE[\hat{v}] = (v(\pi) - \hat{v})^2$

- Importance-Sampling policy evaluation:

  Re-weight reward totals for each trajectory:

  $$IS(\pi, H, \pi_b) := \left(\prod_{t=1}^{L} \frac{\pi(A_t|S_t)}{\pi_b(A_t|S_t)}\right) \times \sum_{t=1}^{L} R_t$$

  Relative likelihood

- Behavior policy gradient: $\theta_{i+1} = \theta_i - \alpha \frac{\partial}{\partial \theta} MSE[IS(\pi, H, \pi_\theta)]$

$$\frac{\partial}{\partial \theta} MSE[IS(\pi, H, \pi_\theta)] = \mathbb{E}_{\pi_\theta}\left[-IS(\pi, H, \pi_\theta)^2 \sum_{t=0}^{L} \frac{\partial}{\partial \theta} \log \pi_\theta(A_t|S_t)\right]$$

- Regression importance sampling: IS with maximum likelihood policy
  1. learn π_b with supervised learning

     $$\pi_{\hat{\theta}} = \underset{\theta}{\operatorname{argmax}} \sum_{(S,A) \in D} \log \pi_\theta(A|S)$$

  2. Importance weight for trajectory H:

     $$IS(\pi, H, \pi_{\hat{\theta}}) = \prod_{t=0}^{L} \frac{\pi_e(A_t|S_t)}{\pi_{\hat{\theta}}(A_t|S_t)} \sum_{t=0}^{L} \gamma^t R_t$$