Challenges of Real-World RL

Gabriel Dulac-Arnold[1], Daniel Mankowitz[*], Todd Hester[*]

[1]Google Research, Brain Team [*]Deepmind

Summary

- Reinforcement learning has proven its worth in numerous artificial domains.
- Many of the recent advances in RL research are hard to leverage in real-world systems, primarily due to rarely satisfied assumptions.
- We propose 9 challenges that must be addressed to bring RL to real-world systems, and discuss current approaches to each of these.
- We propose example domains, as well as a full training/evaluation regime that more closely resembles real-world systems and acts as a more realistic testbed for candidate RL algorithms.

Environments

Cart-Pole Variables: x, θ

Type	Constraint
Static	Limit range: $x_l < x < x_r$
Kinematic	Limit velocity near goal: $\lvert\theta_c - \theta\rvert > \theta_L \vee \dot{\theta} < \dot{\theta}_V$
Dynamic	Limit cart acceleration: $\ddot{x} < A_{\text{max}}$

Walker Variables: $\boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{F}$

Type	Constraint
Static	Limit joint angles: $\boldsymbol{\theta}_L < \boldsymbol{\theta} < \boldsymbol{\theta}_U$ Enforce uprightness: $0 < \boldsymbol{u}_x$
Kinematic	Limit joint velocities: $\max_i \lvert\dot{\boldsymbol{\theta}}_i\rvert < L_{\dot{\theta}}$
Dynamic	Limit foot contact forces: $\boldsymbol{F}_{\text{foot}} < F_{\text{max}}$

Robustness

Env.	Noise	Non-Stationarity
Cart-Pole	Actuator and sensor delays	Track friction increasing with time
Walker	Noisy perception of terrain	Occasionally non-responsive leg actuator
Manipulator	Imprecise proprioception	Changes in gripper friction
Humanoid	Reduced torque on leg actuator	Varying payload CoGs

Manipulator Variables: $\boldsymbol{\theta}, \boldsymbol{F}, \mathcal{M}$

Type	Constraint
Static	Limit joint angles $\boldsymbol{\theta}_L < \boldsymbol{\theta} < \boldsymbol{\theta}_U$ Avoid dynamic obstacles $\mathcal{M} \cap \mathcal{M}_{O,i} = \varnothing$ Avoid self-contact $\mathcal{M} \cap \mathcal{M} = \mathcal{M}$
Kinematic	Limit joint velocities: $\max_i \lvert\dot{\boldsymbol{\theta}}_i\rvert < L_{\dot{\theta}}$
Dynamic	Acceleration Limits: $\max_i \lvert\ddot{\boldsymbol{\theta}}_i\rvert < L_{\ddot{\theta}}$ Limit end effector forces: $\boldsymbol{F}_{\text{EE}} < F_{\text{max}}$

Humanoid Variables: $\boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{F}$

Type	Constraint
Static	Limit joint angles: $\theta_{L,i} < \boldsymbol{\theta}_i < \theta_{U,i}$ Enforce uprightness: $0 < \boldsymbol{u}_x$
Kinematic	Limit joint velocities: $\max_i \lvert\dot{\boldsymbol{\theta}}_i\rvert < L_{\dot{\theta}}$
Dynamic	Limit foot contact forces. $\boldsymbol{F}_{\text{Foot}} < F_{\text{max}}$ Encourage falls on posterior $\boldsymbol{F}_i < F_{\text{max},1} \forall i \in \mathcal{C} \setminus i_{\text{post}}$ $\boldsymbol{F}_{post} < F_{\text{max},2}$

Challenges

Real-world policies need to be:

1. **Off-policy & off-line**: policies need to be trained off-line from the fixed logs of an external behavior policy.
2. **Efficient**: Learning on the real system from limited samples.
3. **Scalable**: policies need to reason with high-dimensional continuous state and action spaces.
4. **Safe**: environments have safety constraints that should never or at least rarely be violated.
5. **Risk-Adverse & Robust**: environments may be partially observable, non-stationary or stochastic and potentially adversarial.
6. **Discerning**: policies need to reason with reward functions that are unspecified, multi-objective, or risk-sensitive.
7. **Explainable**: system operators require explainable policies and actions.
8. **Fast**: Inference must happen in real-time at the control frequency of the system.
9. **Mnemonic**: Policies should be able to handle large and/or unknown delays in the system actuators, sensors, or rewards.

Evaluators

Challenge	Evaluator
Off-line	$J^{start} = R(\texttt{Train}(D_{\pi_B}))$
Efficient	$J^{eff.} = \min \lvert\mathcal{D}_i\rvert$ s.t. $R(\texttt{Train}(D_i)) > R_{\text{min}}$
Safe	$\boldsymbol{J}^{safety}(\pi) = \left(\sum_{i=1}^{T} c_j(s_i, a_i)\right)_{1 \le j \le K} \in \mathbb{R}^K$
Robust	$J^{robust}(\pi) = \frac{1}{K} \sum_{p \in \mathbf{P}} \mathbf{E}^p \left[\sum_{i=1}^{T} r(s_i, a_i)\right]$
Discerning	$\boldsymbol{J}^{multi}(\pi) = \left(\sum_{i=1}^{T_n} r_j(s_i, a_i)\right)_{1 \le j \le K} \in \mathbb{R}^K$

Proposed Framework

- Training performed in Batch-RL (off-line, Initially off-policy)
  - Training from behavior policy logs with varying sizes and policy qualities.
- Training considered as on-line:
  - Use both safety conscious and robustness-enhanced environments.
  - Every interaction should be considered as the real system being run (goes towards efficiency counts and safety constraints).
- Evaluate according to the proposed evaluators, and compare training algorithms according to a multi-dimensional approach.