Randomized Ensembled Double Q-Learning: Learning Fast Without a Model

Xinyue Chen*, Che Wang*, Zijian Zhou*, Keith Ross

New York University/New York University shanghai

NYU

NYU 上海 SHANGHAI 纽约大学

Quick summary of paper:

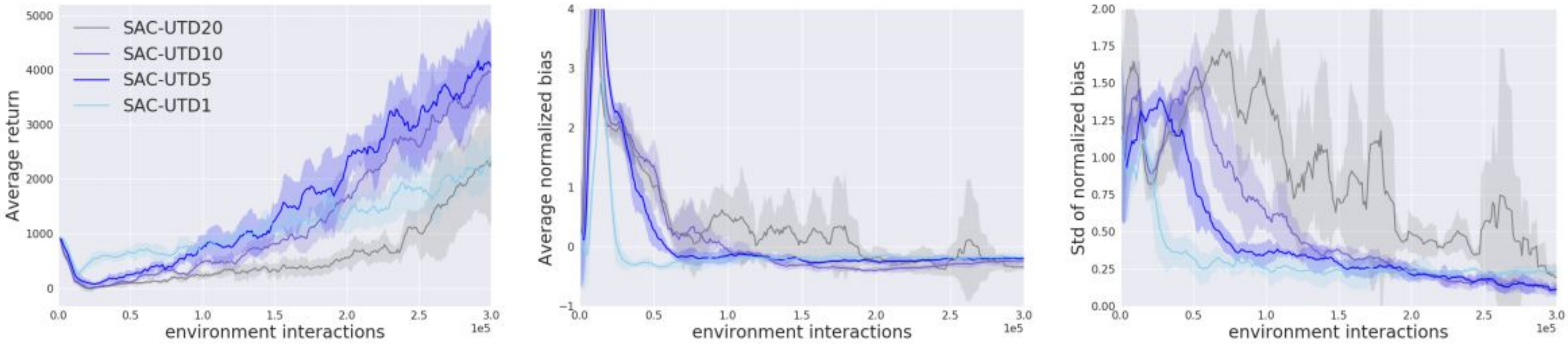
- Randomized Ensembled Double Q-Learning (**REDQ**)
- Uses high update-to-data (**UTD**) ratio, better **bias control**
- **Massive 3-8x better** sample efficiency than SAC baseline
- Model-free, simple, effective, easy to implement
- Clean public **code** with **video tutorial**

Motivation:

1. Update-to-data (UTD) ratio: the number of updates taken compared to the number of actual interactions with the environment.
2. Model-based methods such as MBPO have recently achieved very strong sample efficiency, with a high UTD ratio.
3. For model-free methods, their sample efficiency is much lower and the high UTD scenario has rarely been studied.

Naively increase the UTD: does not work.

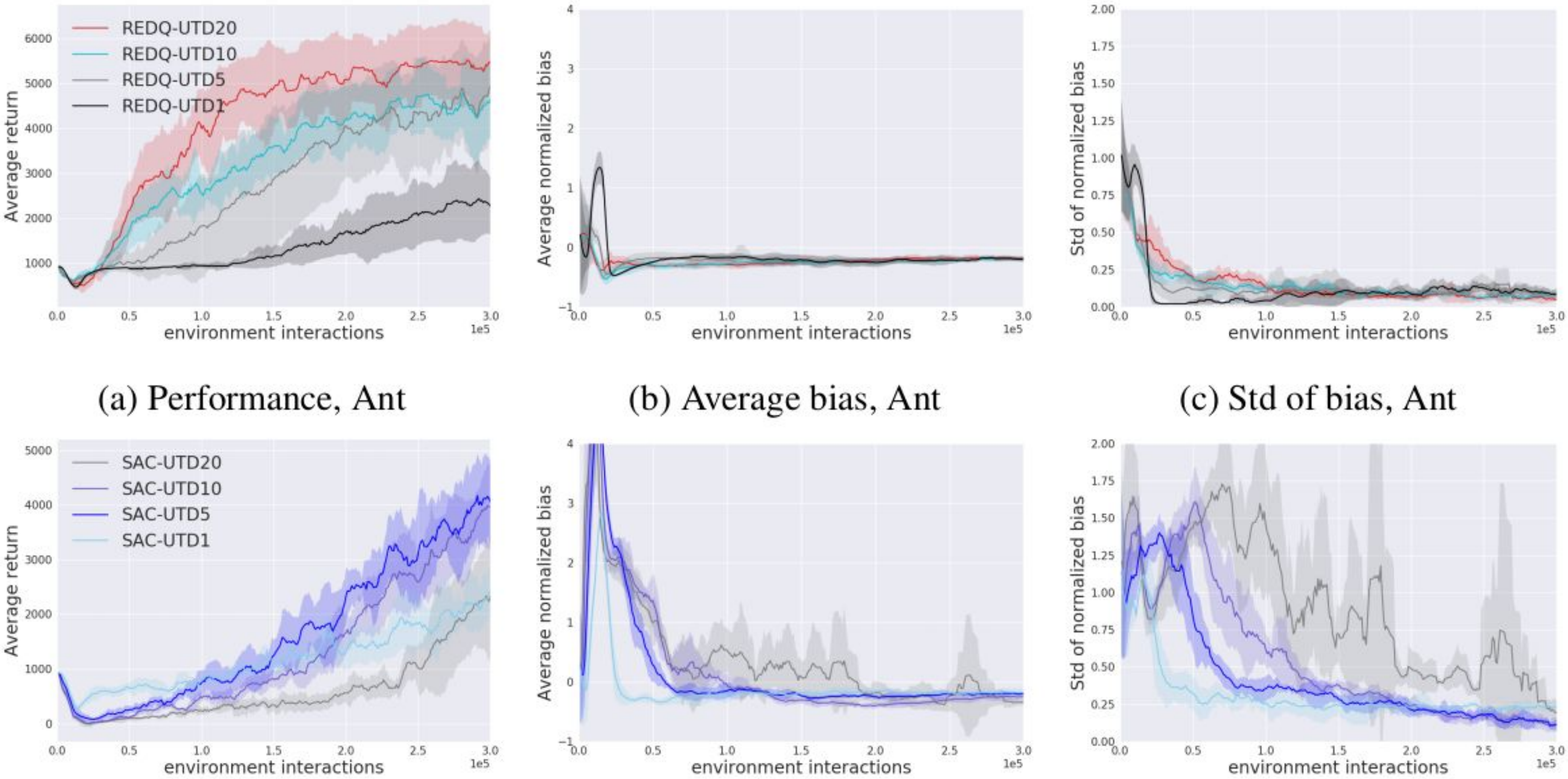
- Higher UTD makes Q value bias higher, more non-uniform (higher std).
- Bias estimated with MC returns.



(d) Performance, Ant

(e) Average bias, Ant

(f) Std of bias, Ant



(a) Performance, Ant

(b) Average bias, Ant

(c) Std of bias, Ant

(d) Performance, Ant

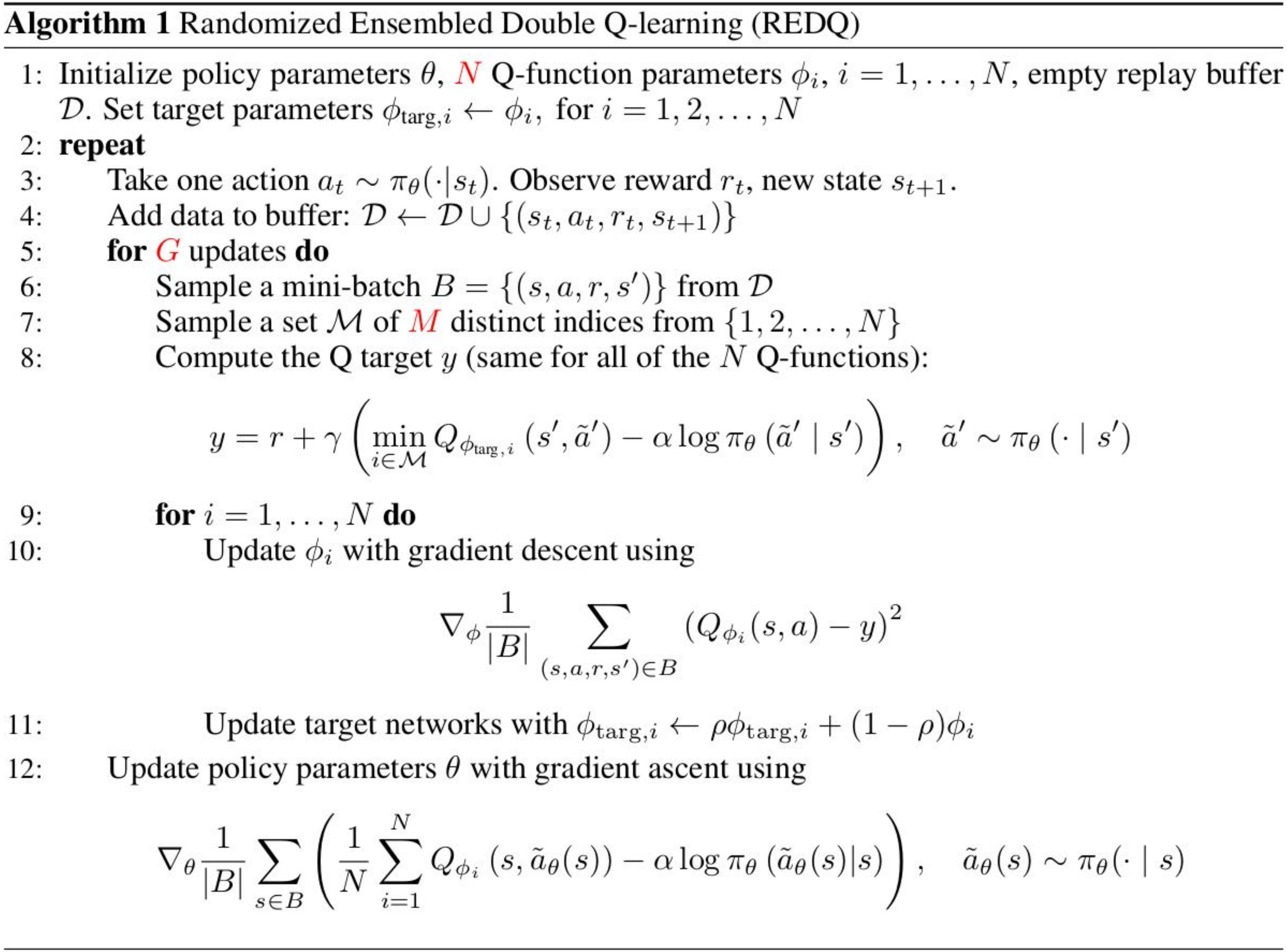
(e) Average bias, Ant

(f) Std of bias, Ant

Summary of REDQ algorithm:

We propose a new model-free deep reinforcement learning (DRL) algorithm called Randomized Ensembled Double Q-Learning (REDQ).

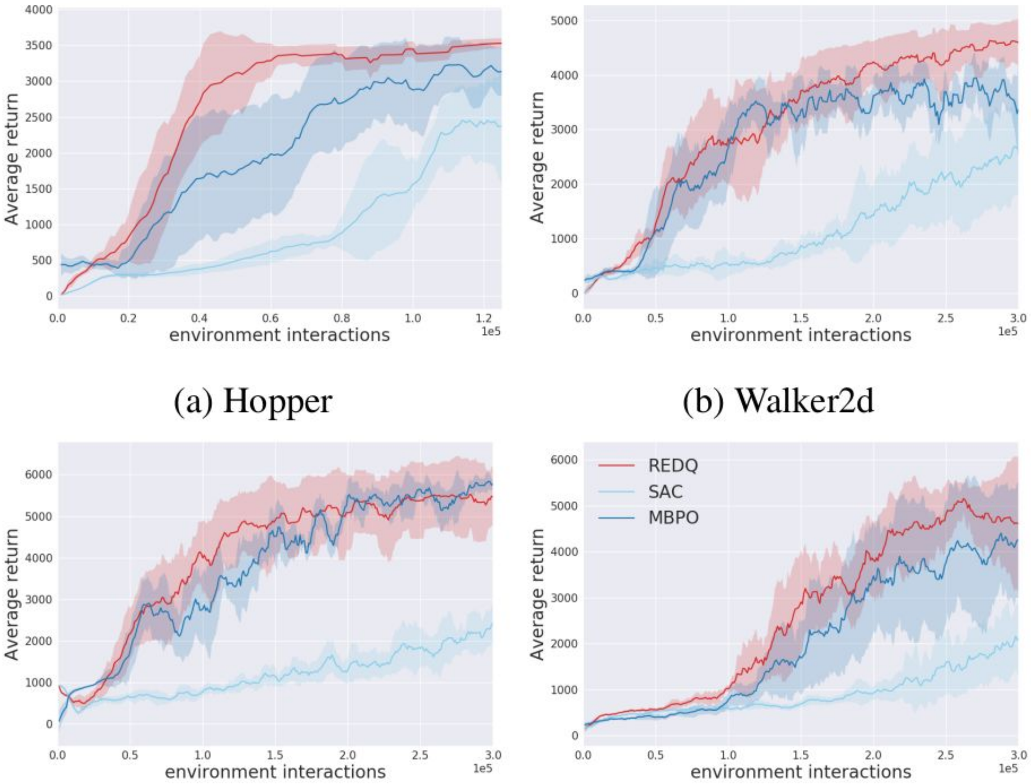
1. REDQ works well under a high update-to-data (UTD) ratio
2. REDQ uses an ensemble of Q networks to control variance
3. REDQ uses in-target minimization across a random subset of Q functions from the ensemble to control bias



Algorithm 1 Randomized Ensembled Double Q-learning (REDQ)

```
Initialize policy parameters $\theta$, $N$ Q-function parameters $\phi_i$, $i = 1, \ldots, N$, empty replay buffer $\mathcal{D}$. Set target parameters $\phi_{\text{targ},i} \leftarrow \phi_i$, for $i = 1, 2, \ldots, N$
repeat
    Take one action $a_t \sim \pi_\theta(\cdot|s_t)$. Observe reward $r_t$, new state $s_{t+1}$.
    Add data to buffer: $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r_t, s_{t+1})\}$
    for $G$ updates do
        Sample a mini-batch $B = \{(s, a, r, s')\}$ from $\mathcal{D}$
        Sample a set $\mathcal{M}$ of $M$ distinct indices from $\{1, 2, \ldots, N\}$
        Compute the Q target $y$ (same for all of the $N$ Q-functions):
            $y = r + \gamma \left( \min_{i \in \mathcal{M}} Q_{\phi_{\text{targ},i}}(s', \tilde{a}') - \alpha \log \pi_\theta(\tilde{a}' \mid s') \right), \quad \tilde{a}' \sim \pi_\theta(\cdot \mid s')$
        for $i = 1, \ldots, N$ do
            Update $\phi_i$ with gradient descent using
                $\nabla_\phi \frac{1}{|B|} \sum_{(s,a,r,s') \in B} \left( Q_{\phi_i}(s, a) - y \right)^2$
            Update target networks with $\phi_{\text{targ},i} \leftarrow \rho \phi_{\text{targ},i} + (1 - \rho)\phi_i$
    Update policy parameters $\theta$ with gradient ascent using
        $\nabla_\theta \frac{1}{|B|} \sum_{s \in B} \left( \frac{1}{N} \sum_{i=1}^{N} Q_{\phi_i}(s, \tilde{a}_\theta(s)) - \alpha \log \pi_\theta(\tilde{a}_\theta(s) | s) \right), \quad \tilde{a}_\theta(s) \sim \pi_\theta(\cdot \mid s)$
```

REDQ (UTD=20) performance compared to MBPO (UTD=20) and SAC (UTD=1):



(a) Hopper

(b) Walker2d

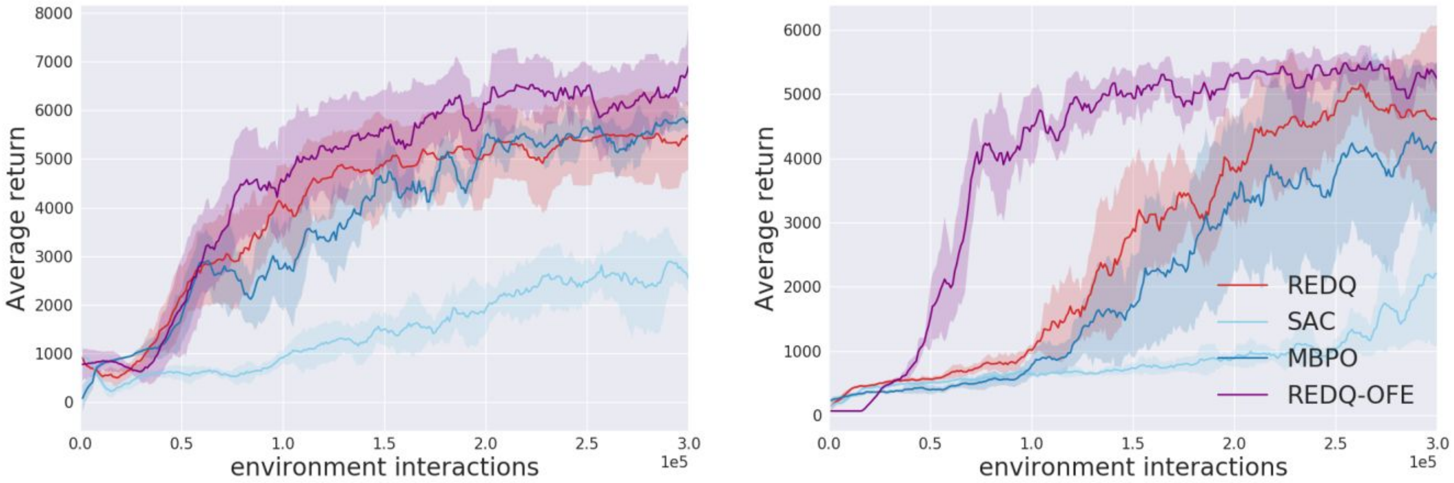
(c) Ant

(d) Humanoid

Results show that:

1. REDQ achieves a substantial 3x-8x better sample efficiency compared to Soft Actor-Critic (SAC).
2. REDQ achieves similar or stronger performance compared to Model-Based Policy Optimization (MBPO). REDQ achieves this with less computation and fewer parameters.
3. When combined with online feature extractor network (OFENet), REDQ greatly outperform MBPO in Ant and Humanoid. (shown next)

REDQ and REDQ-OFE (UTD=20) performance compared to MBPO (UTD=20) and SAC (UTD=1):



(a) Performance, Ant

(b) Performance, Humanoid

Figure 4: Performance of REDQ, REDQ with OFE, and SAC.

Fairness of comparison, consistency with prior work, reproducibility

1. Our MBPO results are obtained with authors’ code with authors’ hyperparameters.
2. We keep all our hyperparameters exactly the same as MBPO whenever possible (UTD, learning rate, batch size, etc.). We also use the same number of datapoints, and the same evaluation protocols.
3. REDQ hyperparameters (N, G, M) are exactly the same for all environments.
4. All REDQ related ablations use the same codebase.
5. Source code and implementation tutorial can be found at: https://github.com/watchernyu/REDQ

Conclusion:

1. We propose REDQ, a model-free method that can achieve strong sample efficiency under high UTD.
2. With extensive experiments we explain why REDQ works so well and discuss several variants. A large number of ablation studies can be found in the paper.
3. We combine REDQ with OFE and show REDQ-OFE can learn extremely fast for the most challenging environments Ant and Humanoid.

Thank you!