Supplementary Information

April 25, 2024

1 algorithms

Algorithm 1 Fix-weight

Input: State S_t, weights $\mathbf{w}$
Initialize: K weight-conditioned Actor-critic networks $\boldsymbol{\pi_i}$ and $\boldsymbol{v^{\pi_i}}$, predetermined sub-space $\mathbf{W}_i$, and memory buffer $\boldsymbol{E}$ size of B.
for $i = 0$ **to** n **do**
  for $t = 1$ **to** B **do**
    $\mathbf{w} \leftarrow$ get pivot weight$(\mathbf{W}_i)$
    $a_t \leftarrow \pi_i(S_t, \mathbf{w})$
    $S_{t+1}, \mathbf{r_t} \leftarrow \mathit{simulator}(a_t)$
    $\mathbf{E} \leftarrow \mathbf{E} \cup \langle S_t, a_t, \mathbf{w}, \mathbf{r_t}, S_{t+1} \rangle$
    $S_t \leftarrow S_{t+1}$
  end for
  sample $\langle S_t, a_t, \mathbf{w}, \mathbf{r_t}, S_{t+1} \rangle \leftarrow \mathbf{E}$
  $\theta \leftarrow \theta + \eta \nabla_\theta \log \pi_\theta(s, a; \mathbf{w}) A^\pi(s_t, a_t; \mathbf{w})$
  $\phi \leftarrow \phi + ||\boldsymbol{V}^{\pi_i}(S; \mathbf{w}) - \boldsymbol{V}^{\pi_i}(S_{t+1}; \mathbf{w})||^2$
  clear $\mathbf{E}$
end for

Algorithm 2 Random-weight

Input: State S_t, weights $\mathbf{w}$
Initialize: K weight-conditioned Actor-critic networks $\boldsymbol{\pi_i}$ and $\boldsymbol{v^{\pi_i}}$, predetermined sub-space $\mathbf{W}_i$, weight change period k, and memory buffer $\boldsymbol{E}$ size of B.
for $i = 0$ **to** K **do**
  for $t = 0$ **to** B **do**
    if $t \mod k = 0$ **then**
      $\mathbf{w}_t \leftarrow$ uniform sample$(\mathbf{W}_i)$
    end if
    $a_t \leftarrow \pi_i(S_t, \mathbf{w}_{\boldsymbol{t}})$
    $S_{t+1}, \mathbf{r_t} \leftarrow \mathit{simulator}(a_t)$
    $\mathbf{E} \leftarrow \mathbf{E} \cup \langle S_t, a_t, \mathbf{w}_t, \mathbf{r_t}, S_{t+1} \rangle$
    $S_t \leftarrow S_{t+1}$
  end for
  sample $\langle S_t, a_t, \mathbf{w}_t, \mathbf{r_t}, S_{t+1} \rangle \leftarrow \mathbf{E}$
  $\theta \leftarrow \theta + \eta \nabla_\theta \log \pi_\theta(s, a; \mathbf{w}_t) A^\pi(s_t, a_t; \mathbf{w}_t)$
  $\phi \leftarrow \phi + ||\boldsymbol{V}^{\pi_i}(S; \mathbf{w}_t) - \boldsymbol{V}^{\pi_i}(S_{t+1}; \mathbf{w}_t)||^2$
  clear $\mathbf{E}$
end for

Algorithm 3 UCB-MOPPO

Input: State S_t, weights $\mathbf{w}$
Initialize: K weight conditioned Actor-critic network $\boldsymbol{\pi_i}$ and $\boldsymbol{v^{\pi_i}}$, predetermined sub-space $\boldsymbol{W_i}$ size of M, current working weight space $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}}$ size of M , each objective has m dimensions, warm-up iterations Q, objective value collection interval in every C iterations, total $K \times m$ surrogate model ψ_θ, and dynamic weight experience pool $\boldsymbol{E}$ size of B.
for $t = 0$ **to** T **do**
    ▶ ***Warm-up Stage***
    for $i = 0$ to n **do**
        $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}} \leftarrow$ get pivot weights($\boldsymbol{W_i}$)
        $\boldsymbol{\pi_i^*} \leftarrow$ Fix Weight Optimisation($\boldsymbol{\pi_i}$,$\widetilde{\boldsymbol{W}}_{\boldsymbol{i}}$)
    end for
    ▶ ***Collect objective value from simulator***
    if $t \bmod C = 0$ **then**
        for $i = 0$ **to** n **do**
            for w in $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}}$ **do**
                $\boldsymbol{V}^{\pi_i} \leftarrow simulator(\pi_i, \mathbf{w})$
            end for
        end for
    end if
    if $t > Q$ **then**
        ▶ ***Construct Training Data for Prediction Model:***
        for $i = 0$ **to** n **do**
            for w in $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}}$ **do**
                for $k = 1$ to $\frac{Q}{C}$ **do**
                    $\delta\mathbf{V}_k^{\pi_i} \leftarrow \boldsymbol{V}_k^{\pi_i} - \boldsymbol{V}_{k-1}^{\pi_i}$
                end for
            end for
        end for
        ▶ ***Update Prediction Model:***
        for $i = 0$ to n **do**
            for $j = 0$ to m **do**
                for w in $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}}$ **do**
                    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}+$ grid search($\psi_{\theta_j}^{\pi_i}$,$\delta\mathbf{V}_k^{\pi_i}$)
                end for
            end for
            ▶ ***Preference Search:***
            for w in $\mathbf{W}_{\boldsymbol{i}}$ **do**
                for w in $\mathbf{W}_{\boldsymbol{i}}$ **do**
                    $\mathbf{V}_{\mathbf{w}}^{\pi_i} \leftarrow simulator(\pi_i, \mathbf{w})$
                    $L \leftarrow append(\boldsymbol{V}_{\mathbf{w}}^{\pi_i})$
                end for
                for $j = 0$ **to** m **do**
                    $\delta\hat{\mathbf{V}}_{\mathbf{w},j}^{\pi_i} \leftarrow \psi_{\theta,j}^{\pi_i}(\mathbf{w}), \mathbf{w} \in W_i$
                    $\delta\widetilde{V}_{\mathbf{w},j}^{\pi_i} \leftarrow \mu(\delta\hat{\mathbf{V}}_{\mathbf{w},j}^{\pi_i}) + \beta(t) * \sigma(\delta\hat{\mathbf{V}}_{\mathbf{w},j}^{\pi_i})$
                    $\widetilde{V}_{\mathbf{w},j}^{\pi_i} \leftarrow V_{\mathbf{w},j}^{\pi_i} + \delta\widetilde{V}_{\mathbf{w},j}^{\pi_i}$
                end for
                $F \leftarrow \{\widetilde{\boldsymbol{V}}_{\mathbf{w}}^{\pi_i}\} \bigcup L \setminus \{\boldsymbol{V}_{\mathbf{w}}^{\pi_i}\}$
                $\mathcal{D} \leftarrow append(\langle \mathbf{w}_i, \mathcal{H}(F) \rangle)$
            end for
            ▶ ***Update Working Preference Pool:***
            $\{\mathbf{w}_i, i \in (0, M)\} \leftarrow$ Sort $\boldsymbol{\omega}$ of $\mathcal{D}$ by $\mathcal{H}(F)$ in descending order
            $\widetilde{\boldsymbol{W}}_{\boldsymbol{i}} \leftarrow \{\mathbf{w}_i, i \in (0, M)\}$
        end for
    end if
end for

2 Benchmark Problems

This section provide detail objective return for each problems. Where C in the following equations is live bonus.

2.1 Swimmer-v2

Observation and action space: $\mathcal{S} \in \mathbb{R}^8, \mathcal{A} \in \mathbb{R}^2$.
The first objective is forward speed in x axis:

$$R_1 = v_x \tag{1}$$

The second objective is energy efficiency:

$$R_2 = 0.3 - 0.15 \sum_i {a_i}^2, \quad a_i \in (-1, 1) \tag{2}$$

2.2 HalfCheetah-v2

Observation and action space: $\mathcal{S} \in \mathbb{R}^{17}, \mathcal{A} \in \mathbb{R}^6$.
The first objective is forward speed in x axis:

$$R_1 = \min(v_x, 4) + C \tag{3}$$

The second objective is energy efficiency:

$$R_2 = 4 - \sum_i {a_i}^2 + C, \quad a_i \in (-1, 1) \tag{4}$$

$$C = 1 \tag{5}$$

2.3 Walker2d-v2

Observation and action space: $\mathcal{S} \in \mathbb{R}^{17}, \mathcal{A} \in \mathbb{R}^6$.
The first objective is forward speed in x axis:

$$R_1 = v_x + C \tag{6}$$

The second objective is energy efficiency:

$$R_2 = 4 - \sum_i {a_i}^2 + C, \quad a_i \in (-1, 1) \tag{7}$$

$$C = 1 \tag{8}$$

2.4 Ant-v2

Observation and action space: $\mathcal{S} \in \mathbb{R}^{27}, \mathcal{A} \in \mathbb{R}^8$.
The first objective is forward speed in x axis:

$$R_1 = v_x + C \tag{9}$$

The second objective is forward in y axis:

$$R_2 = v_y + C \tag{10}$$

$$C = 1 - 0.5 \sum_i a_i^2, \quad a_i \in (-1, 1) \tag{11}$$

2.5 Hopper-v2

Observation and action space: $\mathcal{S} \in \mathbb{R}^{11}, \mathcal{A} \in \mathbb{R}^3$.
The first objective is forward speed in x axis:

$$R_1 = 1.5 v_x + C \tag{12}$$

The second objective is jumping height:

$$R_2 = 12(h - h_{init}) + C \tag{13}$$

$$C = 1 - 2e^{-4} \sum_i a_i^2, \quad a_i \in (-1, 1) \tag{14}$$

2.6 Hopper-v3

Observation and action space: $\mathcal{S} \in \mathbb{R}^{11}, \mathcal{A} \in \mathbb{R}^3$.
The first objective is forward speed in x axis:

$$R_1 = 1.5 v_x + C \tag{15}$$

The second objective is jumping height:

$$R_2 = 12(h - h_{init}) + C \tag{16}$$

The third objective is energy efficiency:

$$R_3 = 4 - \sum_i a_i^2 + C \tag{17}$$

$$C = 1 \tag{18}$$