Algorithm 1 Parallel advantage actor-critic

1: Initialize timestep counter $N = 0$ and network weights θ, θ_v
2: Instantiate set $\boldsymbol{e}$ of n_e environments
3: **repeat**
4:     **for** $t = 1$ to t_{max} **do**
5:         Sample $\boldsymbol{a}_t$ from $\pi(\boldsymbol{a}_t|\boldsymbol{s}_t; \theta)$
6:         Calculate $\boldsymbol{v}_t$ from $V(\boldsymbol{s}_t; \theta_v)$
7:         **parallel for** $i = 1$ to n_e **do**
8:             Perform action $a_{t,i}$ in environment e_i
9:             Observe new state $s_{t+1,i}$ and reward $r_{t+1,i}$
10:         **end parallel for**
11:     **end for**
12:     $\boldsymbol{R}_{t_{\max}+1} = \begin{cases} 0 & \text{for terminal } \boldsymbol{s}_t \\ V(s_{t_{\max}+1}; \theta) & \text{for non-terminal } \boldsymbol{s}_t \end{cases}$
13:     **for** $t = t_{\max}$ down to 1 **do**
14:         $\boldsymbol{R}_t = \boldsymbol{r}_t + \gamma \boldsymbol{R}_{t+1}$
15:     **end for**
16:     $d\theta = \frac{1}{n_e \cdot t_{max}} \sum_{i=1}^{n_e} \sum_{t=1}^{t_{max}} (R_{t,i} - v_{t,i}) \nabla_\theta \log \pi(a_{t,i}|s_{t,i}; \theta) + \beta \nabla_\theta H(\pi(s_{e,t}; \theta))$
17:     $d\theta_v = \frac{1}{n_e \cdot t_{max}} \sum_{i=1}^{n_e} \sum_{t=1}^{t_{max}} \nabla_{\theta_v} (R_{t,i} - V(s_{t,i}; \theta_v))^2$
18:     Update θ using $d\theta$ and θ_v using $d\theta_v$.
19:     $N \leftarrow N + n_e \cdot t_{\max}$
20: **until** $N \geq N_{max}$