Preference-based Evolutionary Direct Policy Search

Róbert Busa-Fekete[1,2] Balázs Szörényi[2,3] Paul Weng[4] Weiwei Cheng[1] Eyke Hüllermeier[1]

[1]Computational Intelligence Group, Philipps University Marburg, Marburg, Germany
[2]Research Group on AI, Hungarian Acad. Sci. and Univ. of Szeged, Szeged, Hungary
[3]INRIA Lille - Nord Europe, SequeL project, Villeneuve d'Ascq, France
[4]Laboratory of Computer Science of Paris 6, University Pierre and Marie Curie, Paris, France

Philipps Universität Marburg

Introduction

- We propose an efficient policy search method for preference-based reinforcement learning
- Evolutionary Direct Policy Search (EDPS) [1]
- Assume a parametric policy space

$$\Pi = \{\pi_\Theta \,|\, \Theta \in \mathbb{R}^p\}$$

- Target function is the *expected total reward*

$$\rho_\pi = \mathop{\mathbb{E}}_{\mathrm{h}\sim \mathrm{P}_\pi}[V(\mathrm{h})]$$

that is estimated based on **rollouts**

- Optimize it by using an *Evolution strategy* (ES), such as CMA-ES [2]
  - If the number of rollouts is too large, the learning process gets slow
  - If the number of rollouts is too small, the ranking over the offsprings is not reliable enough
- Adaptive control of the number of rollouts using racing algorithms

Algorithm 1 EDPS $(\mathcal{M}, \mu, \lambda, n_{\max}, \delta)$

Initialization: select an initial parameter vector $\Omega^{(0)}$ and an initial set of candidate solutions $\Theta_1^{(0)}, \ldots, \Theta_\mu^{(0)}$, $\sigma^{(0)}$ is the identity permutation

$t = 0$

repeat

  $t = t + 1$

  for $\ell = 1, \ldots, \lambda$ **do** ▷ Sample new solutions

    $\Theta_\ell^{(t)} \sim F(\Omega^{(t-1)}, \Theta^{(t-1)}_{\sigma^{(t-1)}(1)}, \ldots, \Theta^{(t-1)}_{\sigma^{(t-1)}(\mu)})$

  end for

  $\sigma^{(t)} =$ **Racing** $\left(\mathcal{M}, \pi_{\Theta_1^{(t)}}, \ldots, \pi_{\Theta_\lambda^{(t)}}, \mu, n_{\max}, \delta\right)$

  $\Omega^{(t)} =$ **Update**$(\Omega^{(t-1)}, \Theta^{(t)}_{\sigma^{(t)}(1)}, \ldots, \Theta^{(t)}_{\sigma^{(t)}(\mu)})$

until Stopping criterion fulfilled

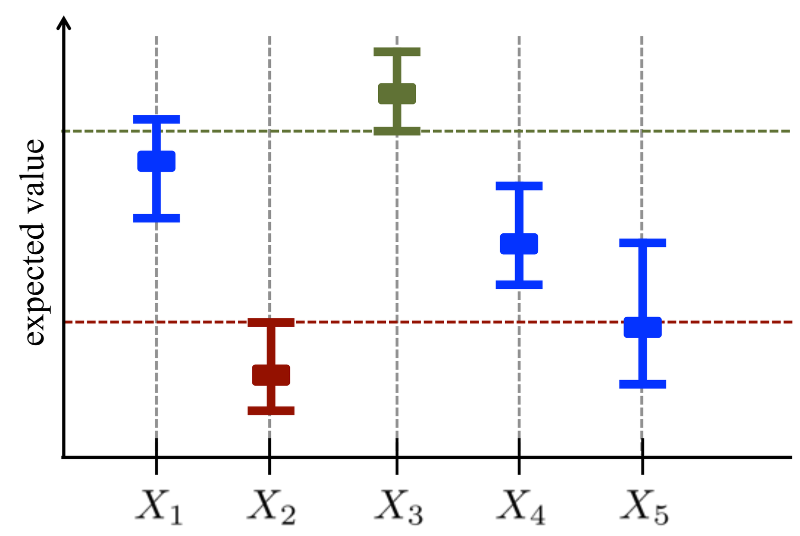
Return $\pi_{\Theta_1^{(t)}}$

Value-based Racing Algorithm [3]

- $X_1, \ldots, X_K$ are random variables with unknown distribution functions $\mathrm{P}_{X_1}, \ldots, \mathrm{P}_{X_K}$ and finite expected values $\mu_i = \int x d\mathrm{P}_{X_i}(x)$
- We solve the following optimization task with high probability

$$\sum_{i\in I}\sum_{j\neq i} \mathbb{I}\{\mu_j < \mu_i\} \longrightarrow \max_{I\subseteq[K]:\,|I|=\kappa}$$

- Hoeffding bound: $\mu_i \in \left[\widehat{\mu}_i - \sqrt{\frac{1}{2n_i}\log\frac{2}{\delta}}, \widehat{\mu}_i + \sqrt{\frac{1}{2n_i}\log\frac{2}{\delta}}\right]$ with probability at least $1-\delta$



Preference-based Reinforcement Learning [4,5]

- Rollout: generating a history $\mathrm{h} \in \mathcal{H}^{(T)}$ by following a policy π for a given MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathrm{P}, r)$
- Assumption: preference relation $\prec$ on $\mathcal{H}^{(T)}$
- Prerequisite: "lifting" of the preference relation $\prec$ on $\mathcal{H}^{(T)}$ to a preference relation $\ll$ on the space of policies Π
- Each policy generates a distribution over the histories $\mathcal{H}^{(T)}$
- We can associate policies with random variables X
- Decision model ($\ll$):

$$X \ll Y \text{ if and only if } \mathrm{P}(Y \prec X) < \mathrm{P}(X \prec Y)$$

- Preferential cycles: $X_1 \ll X_2$, $X_2 \ll X_3$, $X_3 \ll X_1$

Preference-based EDPS

- Evolution Strategies only need a ranking over the candidate solutions to update the parameters of $F(\cdot)$ (distribution over the search space)
- Let's race based on preferences!! $\Rightarrow$ Preference-based EDPS
- Resolving the preferential cycles by using *Copeland relation*:

$$X_i \ll_C X_j \Leftrightarrow d_i < d_j, \quad \text{where} \quad d_i = \#\{k : X_k \ll X_i, X_k \in \mathcal{X}\}$$

- We solve the following optimization task with high probability

$$\sum_{i\in I}\sum_{j\neq i} \mathbb{I}\{X_j \ll X_i\} \longrightarrow \max_{I\subseteq[K]:\,|I|=\kappa}$$

- We need an *efficient estimator* of $S(X_i, X_j) = \mathrm{P}(X_i \prec X_j)$
- A two-sample U-statistic called the *Mann-Whitney U-statistic*

$$\widehat{s}_{i,j} = \widehat{S}(\mathrm{X}_i, \mathrm{X}_j) =$$

$$= \frac{1}{n^2}\sum_{\ell=1}^{n}\sum_{\ell'=1}^{n}\left[\mathbb{I}\{x_i^\ell \prec x_j^{\ell'}\} + \frac{1}{2}[\mathbb{I}\{x_i^\ell \sim x_j^{\ell'}\} + \mathbb{I}\{x_i^\ell \perp x_j^{\ell'}\}]\right]$$

where $\mathrm{X}_i = \{x_i^1, \ldots, x_i^n\} \sim X_i$ and $\mathrm{X}_j = \{x_j^1, \ldots, x_j^n\} \sim X_j$

- Hoeffding, 1963, §5b: For any $\epsilon > 0$, using the notations introduced above,

$$\mathrm{P}\left(\left|\widehat{S}(\mathrm{X}, \mathrm{Y}) - S(X, Y)\right| \geq \epsilon\right) \leq 2\exp(-2n\epsilon^2) \ .$$

Preference-based Racing Algorithm

1. Input: $X_1, \ldots, X_K, \kappa, n_{\max}, \delta$
2. Iteratively sample $X_1, \ldots, X_K$
3. Calculate $\widehat{s}_{i,j}$ for all $1 \leq i, j \leq K$
4. and their confidence intervals as $[\widehat{s}_{i,j} - c_{i,j}, \widehat{s}_{i,j} + c_{i,j}]$ where

$$c_{i,j} = \sqrt{\frac{1}{2n}\log\frac{2K^2 n_{\max}}{\delta}}$$

5. When can we stop sampling an option?
   - Number of options that are beaten by i so far: $z_i = |\{j : u_{i,j} < 1/2, j \neq i\}|$
   - Number of options that beat i so far: $o_i = |\{j : \ell_{i,j} > 1/2, j \neq i\}|$
   - $C = \{i : K - \kappa < |\{j : K - z_j < o_i\}|\}$
   - $D = \{i : \kappa < |\{j : K - o_j < z_i\}|\}$
6. If $(i, j \in C \cup D) \vee (1/2 \notin [\ell_{i,j}, u_{i,j}])$ then do not update $\widehat{s}_{i,j}$ any more

Medical Experiments

- Medical treatment design for cancer clinical trials
- State $s = (S, X)$ describes the health condition of the patient: S is the tumor size and X the level of toxicity
- Action is the dosage level $a \in [0, 1]$
- A history h represents a treatment of a virtual patient
  1. $\mathrm{h}' \preceq \mathrm{h}$ if the patient survives in h but not in h', and both histories are incomparable ($\mathrm{h}' \perp \mathrm{h}$) if the patient does neither survive in h' nor in h.
  2. Otherwise, preference depends on the worst wellness of the patient and the final tumor size: $\mathrm{h}' \preceq \mathrm{h}$ if (and only if) $C_X \leq C'_X$ and $C_S \leq C'_S$ where C_X and C'_X denote the *maximal* toxicity during the whole treatment
  3. Pareto dominance

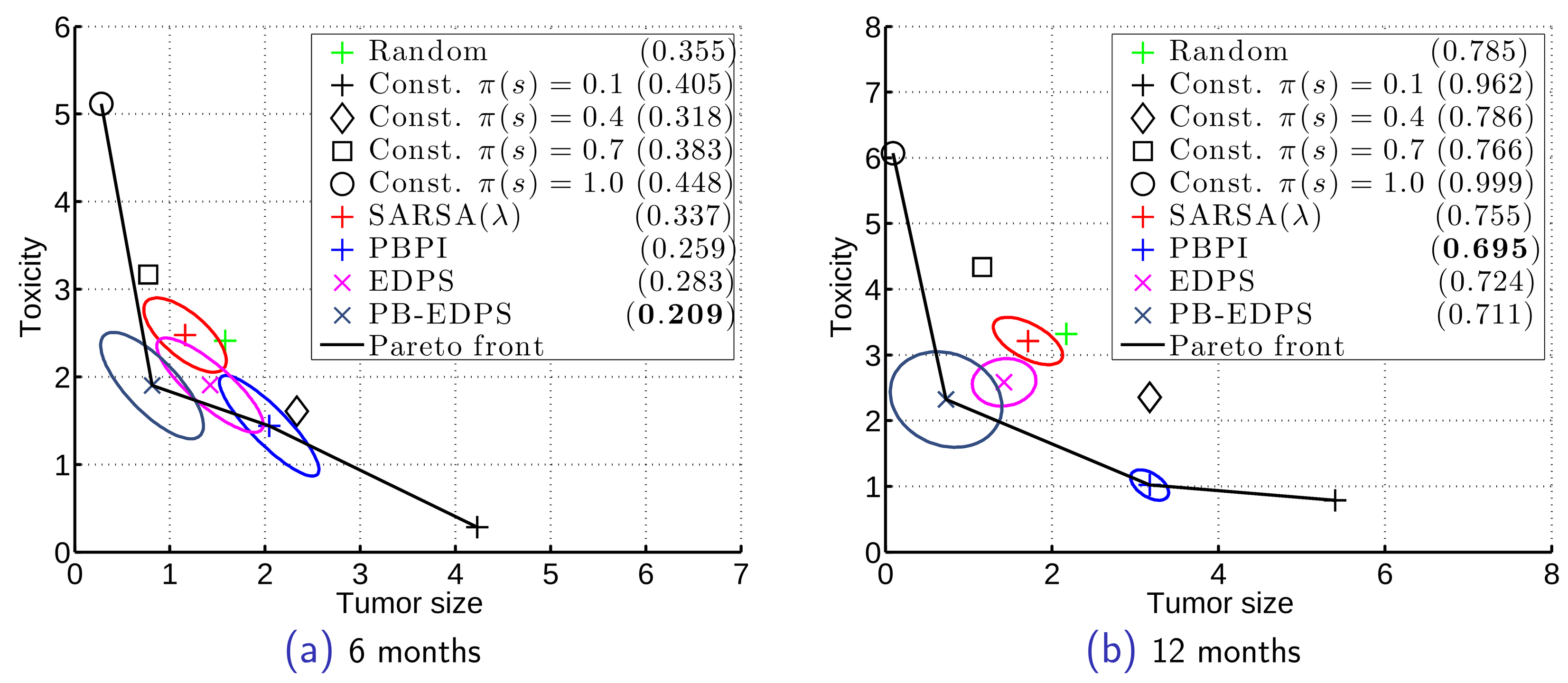


Illustration of patient status under different treatment policies. On the x-axis is the tumor size after 6 (a) and 12 (b) months, on the y-axis the highest toxicity during the treatment. The death rates are shown in parentheses at the upper right corner.

References

[1] Heidrich-Meisner, V., Igel, C.: Hoeffding and Bernstein races for selecting policies in evolutionary direct policy search. ICML, pp. 401–408 (2009)
[2] Hansen, N., Kern, S.: Evaluating the CMA evolution strategy on multimodal test functions. PPSN VIII, pp. 282–291 (2004)
[3] Maron, O., Moore, A.: Hoeffding races: accelerating model selection search for classification and function approximation. NIPS, pp. 59–66 (1994)
[4] Fürnkranz, J., Hüllermeier, E., Cheng, W., Park, S.: Preference-based reinforcement learning: a formal framework and a policy iteration algorithm. Machine Learning **89**(1-2), 123–156 (2012)
[5] Akrour, R., Schoenauer, M., Sebag, M.: Preference-based policy learning. ECML/PKDD, pp. 12–27 (2011)