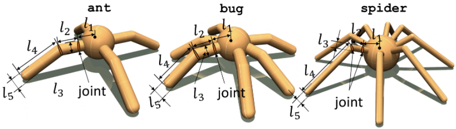
Evolution + deep RL

Evolving something else

Olivier Sigaud

Sorbonne Université
http://people.isir.upmc.fr/sigaud

What else can we evolve?



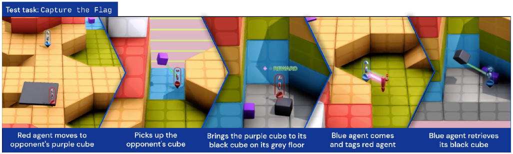
- The environment: POET and many others
- The agent's body (it is part of the environment): DERL and many others
- The hyper-parameters of the algorithm (PBT)
- We focus on the latter

Wang, R., Lehman, J., Clune, J., and Stanley, K. O. (2019) POET: open-ended coevolution of environments and their optimized solutions. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 142–151

Luck, K. S., Amor, H. B., and Calandra, R. (2020) Data-efficient co-adaptation of morphology and behaviour with deep reinforcement learning. In *Conference on Robot Learning*, pages 854–869. PMLR

Park, J. H. and Lee, K. H. (2021) Computational design of modular robots based on genetic algorithm and reinforcement learning. *Symmetry*, 13(3):471

Introduction to PBT

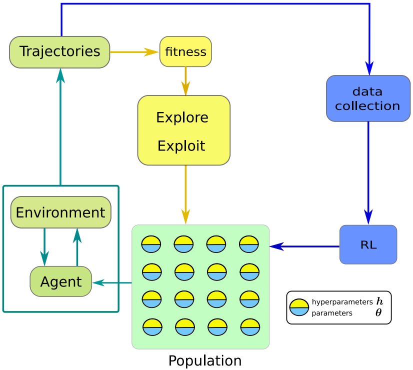


- Hyper-parameter search is crucial in Deep RL
- Population-Based Training (PBT) provides an efficient solution to this problem
- It has been used in several notorious applications of Deep RL (starcraft, oel)
- We note $\mathbf{h}$ the hyper-parameter vector and $\boldsymbol{\theta}$ the parameter vector

Jaderberg, M., Czarnecki, W. M., Dunning, I., Marris, L., Lever, G., Castaneda, A. G., Beattie, C., Rabinowitz, N. C., Morcos, A. S., Ruderman, A., et al. (2019) Human-level performance in 3D multiplayer games with population-based reinforcement learning. *Science*, 364(6443), 859–865

Stooke, A., Mahajan, A., Barros, C., Deck, C., Bauer, J., Sygnowski, J., Trebacz, M., Jaderberg, M., Mathieu, M., et al. (2021) Open-ended learning leads to generally capable agents. *arXiv preprint arXiv:2107.12808*

The PBT architecture



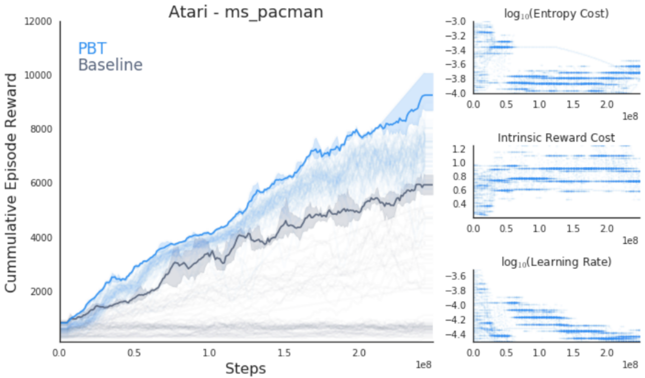
- Applies to policy parameters and hyper-parameters
- A very efficient hyper-parameter tuning approach given a large enough population
- Hyper-parameters evolve along training
- The variation-selection operators (**Exploit**, **Explore**) could be improved

Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., et al. (2017) Population-based training of neural networks. *arXiv preprint arXiv:1711.09846*

Implementation of explore/exploit

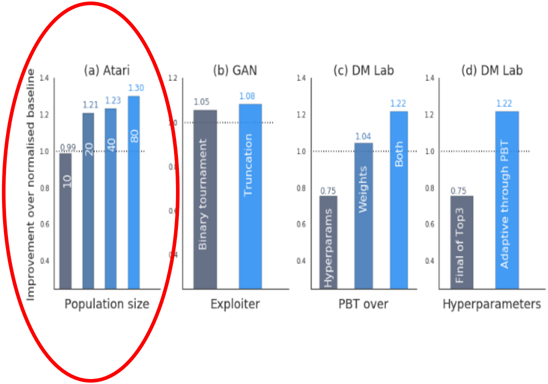
- Two Explore strategies
  - **Perturb:** each hyper-parameter independently is randomly perturbed by a factor of 1.2 or 0.8.
  - **Resample:** each hyper-parameter is resampled from the original prior distribution
  - They seem to only use Perturb
- Two Exploit strategies
  - **T-test selection:** Uniformly sample another agent that replaces the current agent if its last 10 episodic rewards are better using Welch's T-test.
  - **Truncation selection:** Rank all agents, if the current agent is in the bottom K% replace by one in the top K% (K=20 or 5).

Variation of hyper-parameters over time



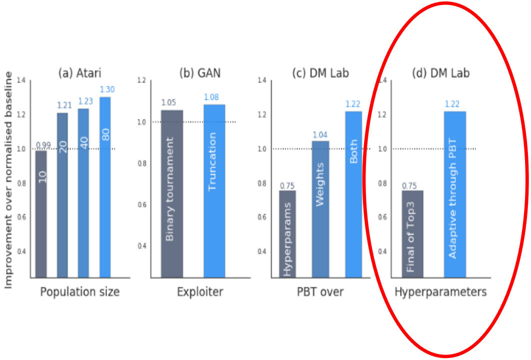
- We can see the $\mathbf{h}$ drifting over time
- Does not converge to a single value

Effect of population size



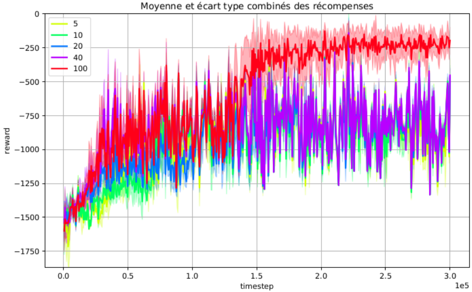
- A too small population results in suboptimal performance
- Adding more agents improves less and less
- More agents means more samples. Should compare at constant budget!

Role of adaptivity



- Measure performance with the fixed final hyper-params
- The fact that $\mathbf{h}$ changes over time is beneficial to performance

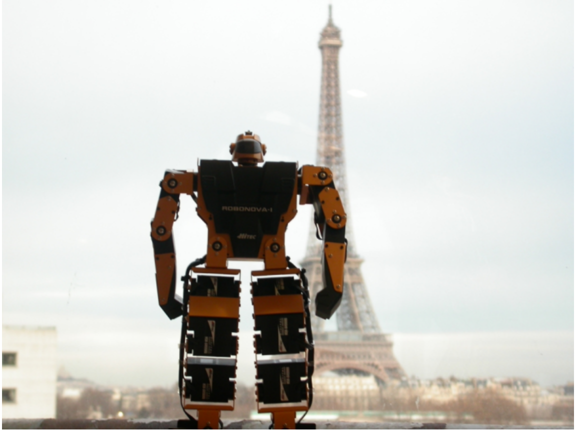
A PBT example



- Not convincing with a small population
- A larger population can find the right hyper-parameters
- Complex architecture, need for large computational resources
- The evolution part is very naive, could be much improved

Pierrot, T. and Flajolet, A. (2023) Evolving populations of diverse RL agents with Map-Elites. *arXiv preprint arXiv:2303.12803* (ICLR 2023)

Any question?



Send mail to: Olivier.Sigaud@isir.upmc.fr

Jaderberg, M., Czarnecki, W. M., Dunning, I., Marris, L., Lever, G., Castaneda, A. G., Beattie, C., Rabinowitz, N. C., Morcos, A. S., Ruderman, A., et al. (2019).
Human-level performance in 3d multiplayer games with population-based reinforcement learning.
Science, 364(6443):859–865.

Jaderberg, M., Dalibard, V., Osindero, S., Czarnecki, W. M., Donahue, J., Razavi, A., Vinyals, O., Green, T., Dunning, I., Simonyan, K., et al. (2017).
Population-based training of neural networks.
arXiv preprint arXiv:1711.09846.

Luck, K. S., Amor, H. B., and Calandra, R. (2020).
Data-efficient co-adaptation of morphology and behaviour with deep reinforcement learning.
In *Conference on Robot Learning*, pages 854–869. PMLR.

Park, J. H. and Lee, K. H. (2021).
Computational design of modular robots based on genetic algorithm and reinforcement learning.
Symmetry, 13(3):471.

Pierrot, T. and Flajolet, A. (2023).
Evolving populations of diverse RL agents with map-elites.
arXiv preprint arXiv:2303.12803.

Stooke, A., Mahajan, A., Barros, C., Deck, C., Bauer, J., Sygnowski, J., Trebacz, M., Jaderberg, M., Mathieu, M., et al. (2021).
Open-ended learning leads to generally capable agents.
arXiv preprint arXiv:2107.12808.

Wang, R., Lehman, J., Clune, J., and Stanley, K. O. (2019).
POET: open-ended coevolution of environments and their optimized solutions.
In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 142–151.