On the Importance of Hyperparameter Optimization in Model-based Reinforcement Learning

Baohe Zhang, Raghu Rajan, Luis Pineda,
Nathan Lambert, André Biedenkapp, Kurtland Chua,
Frank Hutter, Roberto Calandra

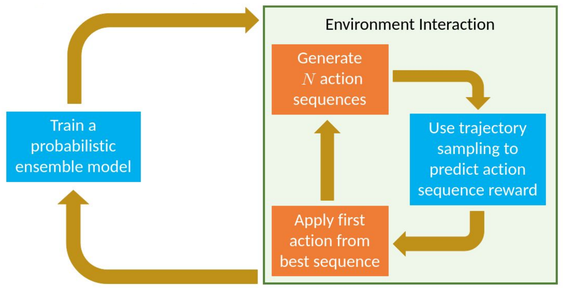
universität freiburg
Meta AI · Berkeley UNIVERSITY OF CALIFORNIA · PRINCETON UNIVERSITY

Motivation

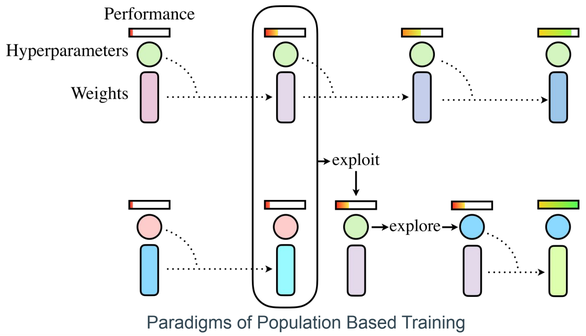
- RL, in general, is very sensitive to hyperparameters. (Henderson et al., 2018).
- Model-based RL (MBRL) training is a non-stationary process and involves model fitting and planning, which increases the complexity compared to traditional RL.
- Research questions:
  - How to find a good hyperparameter configuration without manually tuning it?
  - How to demonstrate and address the non-stationarity of MBRL?

Background

- MBRL: We use Probabilistic ensembles with trajectory sampling (PETS) (Chua et al., 2018) as our MBRL algorithm because of its state-of-the-art performance.



- Hyperparameter Optimization (HPO):
  - Random Search and Hyperband (Li et al., 2017) for static tuning.
  - Population Based Training (PBT) (Jaderberg et al., 2017) and PBT with Backtracking (PBT-BT) for dynamic tuning.

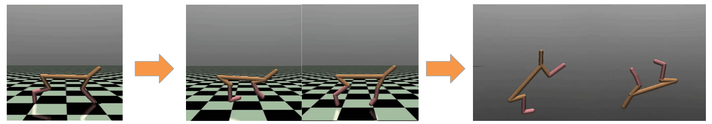


Paradigms of Population Based Training

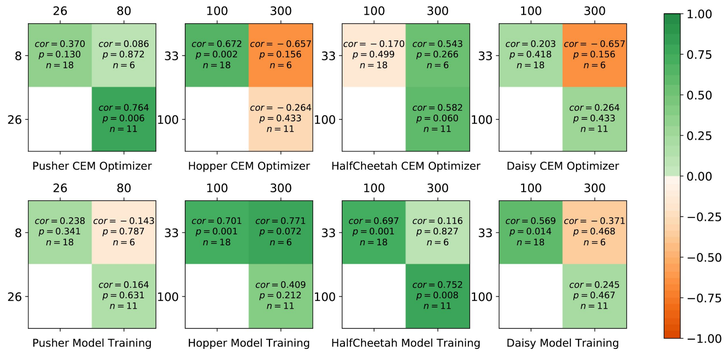
Experimental Results

Need for Dynamic Tuning

Intuition: For a given task, different hyperparameters are likely to be optimal for policies learnt at different stages of training.

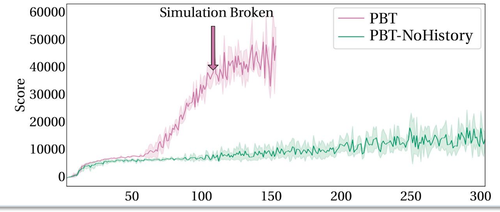


Spearman rank correlations of hyperparameter configurations across different training budgets are generally very small or even negative. This indicates that different configurations may be required in different stages of a task.

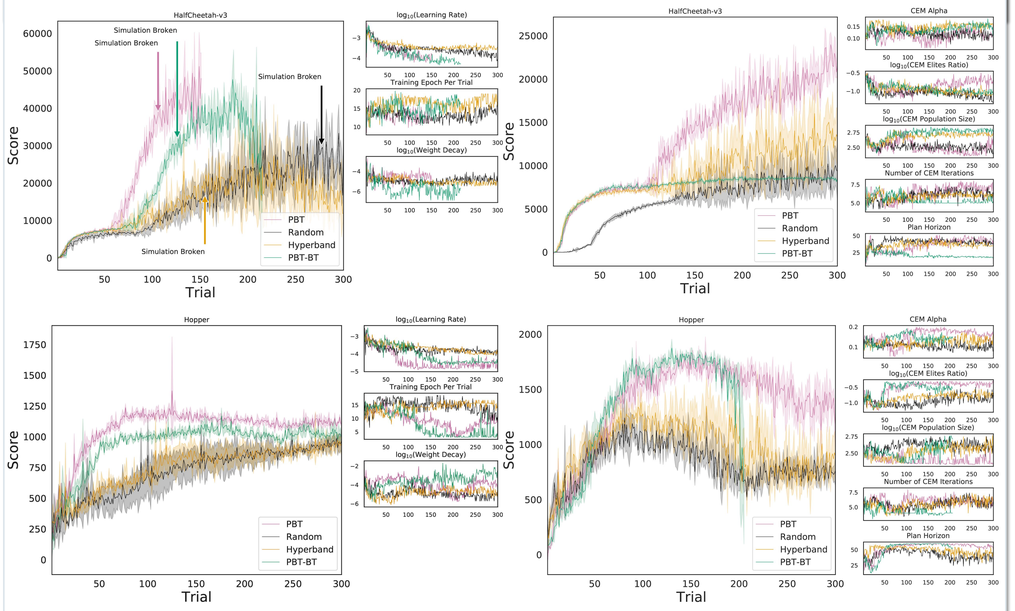


The Importance of History

In the original PBT, the Replay Buffer was not copied during the exploitation step. We show that copying the history/training data is an essential step to applying PBT on MBRL algorithms



The Importance of HPO



We evaluate four different HPO methods on Mujoco Environments. The curves show the average returns and hyperparameter configurations of the top 5 members in the search population.

- Dynamic tuning methods are able to learn a learning rate schedule with a decaying pattern without any predefined scheduler. Static methods are able to find the schedule, but can not utilize this finding.
- Dynamic tuning methods outperform static tuning in most of the environments when tuning sets of hyperparameters separately. (More results are available in the paper)
- With HPO, we manage to break the simulation in HalfCheetah.
- The importance of the same hyperparameters in different environments are also different, which again shows the need for HPO.
- The *planning horizon* has an increasing trend, which supports that longer planning may be better when the uncertainty in the model is lower.

Conclusion

- HPO methods can significantly improve the performance of MBRL.
- Dynamic tuning can be better than static tuning with regard to the final reward by addressing the non-stationarity of MBRL.
- The history plays an essential role when applying dynamic tuning in MBRL.

References

Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D. (2018). "Deep reinforcement learning that matters." In: AAAI Conference on Artificial Intelligence

Chua, K., Calandra, R., McAllister, R., and Levine, S. (2018). Deep reinforcement learning in a handful of trials using probabilistic dynamics models. In: Advances in Neural Information Processing Systems, pages 4754–4765.

Li, L., Jamieson, K., DeSalvo, G., Rostamizadeh, A., and Talwalkar, A. (2017). Hyperband: A novel bandit-based approach to hyperparameter optimization. In: The Journal of Machine Learning Research (JMLR), 18(1):6765–6816.

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. In: arXiv preprint arXiv:1711.09846.

Paper Video

AISTATS