Reinforcement Learning for Batch Bioprocess Optimisation

Panagiotis Petsagkourakis [a]
Ilya Orson Sandoval [b]
Eric Bradford [c]

Code available at:
https://gitlab.com/IlyaOrson/BatchReactor

Introduction:

Bioprocesses have recently received attention to produce clean and sustainable alternatives to fossil-based materials. However, they are generally difficult to optimize due to their unsteady-state operation modes and stochastic behaviours. Furthermore, biological systems are highly complex, therefore plant-model mismatch is often present.

In this work we leverage a model-free Reinforcement Learning optimisation strategy. We apply the Policy Gradient method to tune a control policy parametrized by a recurrent neural network. We assume that a preliminary model of the process is available, which is exploited to obtain an initial optimal control policy. Subsequently, this policy is updated based on a variation of the starting model, with adequate disturbance, to simulate the plan-model mismatch.

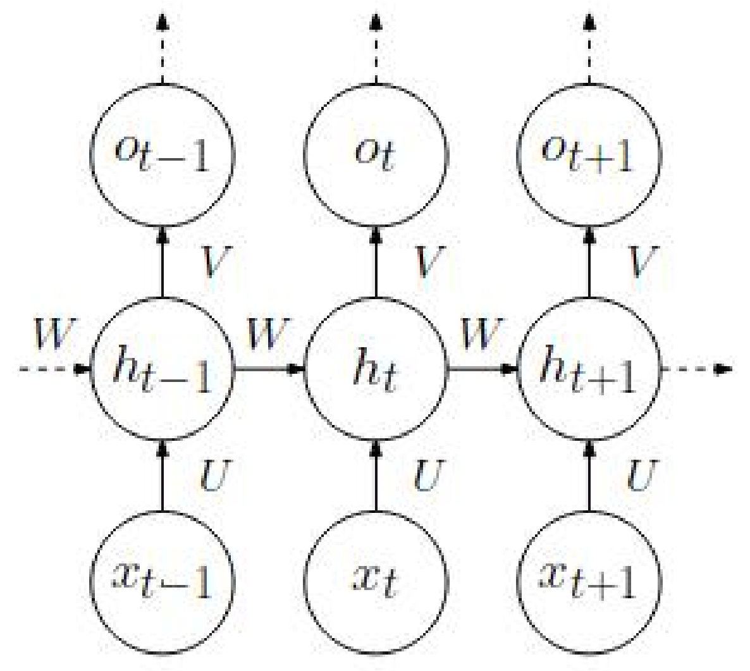
Policy gradient with LSTM policy:

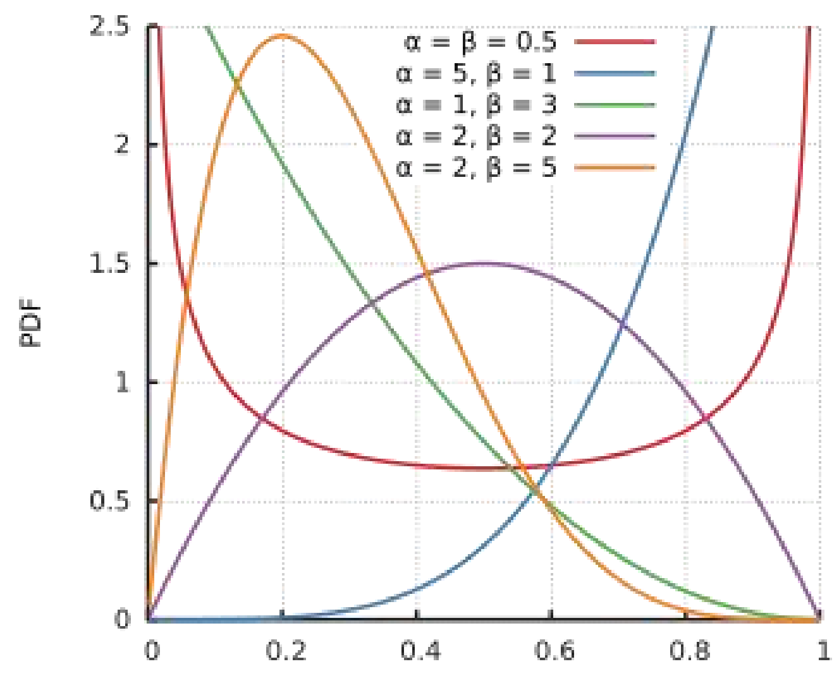
The goal is to optimise the final byproduct of the reaction (y_2) after the controled evolution of the system through the (discretized) time lapse [0, 1]. The bioprocess is modelled by two variations of an ODE system with contrained controls $U_1(t)$, $U_2(t)$ in [0, 5]:

$$\dot{y_1} = -(U_1 + aU_1^2)y_1 + dU_2 \qquad \dot{y_1} = -(U_1 + aU_2^2)y_1 + dU_2 y_2/(y_1 + y_2)$$

$$\dot{y_2} = (bU_1 - cU_2)y_1 \qquad \dot{y_2} = (bU_1 - cU_2)y_1$$

The controls are sampled from a Beta distribution, shifted to the allowed domain of the controls, which mean and variance are predicted by a parametrized policy (π_θ) that ingests the current state of the system ($x=(y_1, y_2)$). In our case, the policy is a recurrent neural network (the LSTM variant) that also takes into account the past states of the policy. This has two main benefits: a better control due to complete trajectory consideration and faster adaptability after model transitions (model-plan mismatch simulation) by transfer learning techniques. We freeze all but the latest layers of the policy resulting in faster training.





The policy gradient theorem gives a way to estimate the gradient of the policy independent of the as an expectation over multiple samples of evolution (τ) following the current policy $\pi_{\theta.}$ A gradient ascent technique is used to adjust the parameters (θ) to maximise the byproduct of the system ($J(\tau) = y_2(t=1)$) as follows:

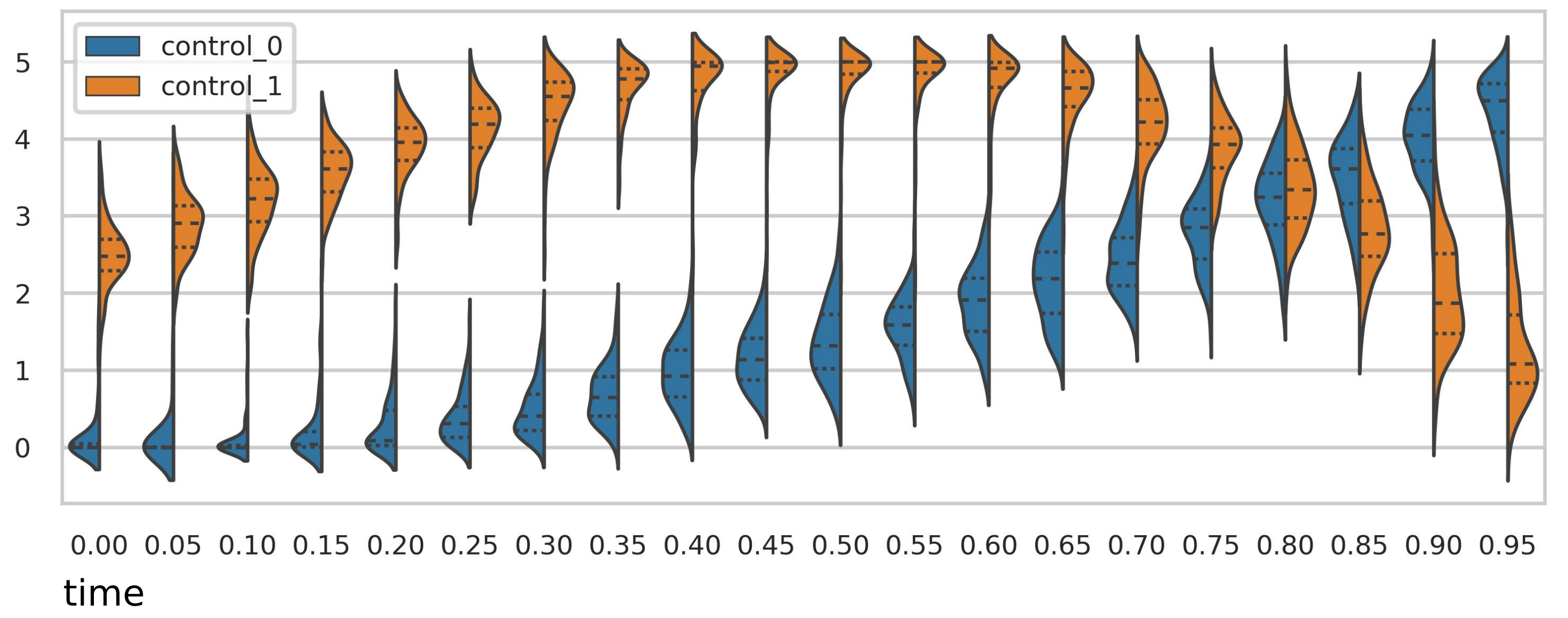
$$p(\tau|\theta) = p(x_0) \prod_{t=0}^{T-1} \pi_\theta(u_t|x_t) p(x_{t+1}|x_t, u_t)$$

$$\nabla_\theta \langle J \rangle_{\tau \sim \pi_\theta} = \mathbb{E} \left[\nabla_\theta \log p(\tau|\theta) \left(J(\tau) - \langle J \rangle_{\tau \sim \pi_\theta} \right) \right]$$

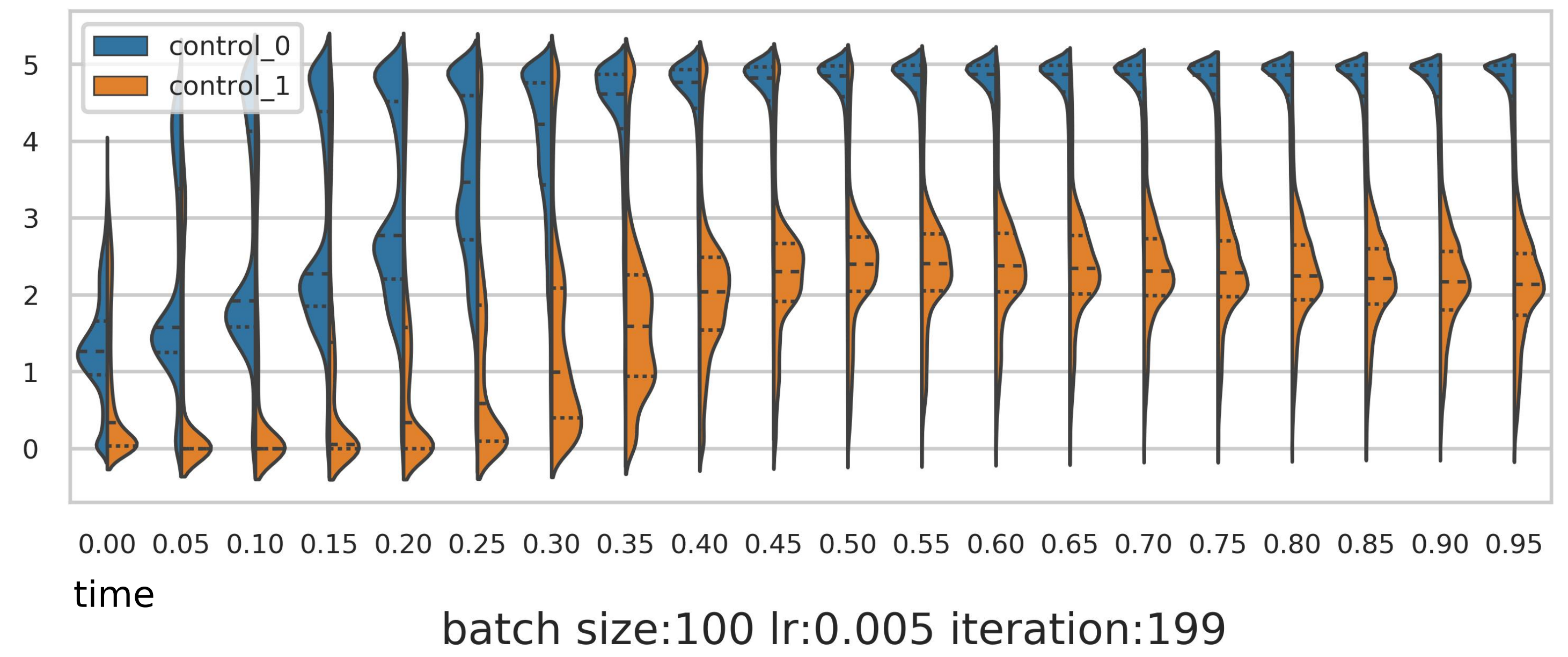
$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta \hat{J}$$

Here a simple mean baseline $\langle J \rangle_{\tau \sim \pi_\theta}$ is added to reduce the variance of the estimator, accelerating the convergece of the algorithm.

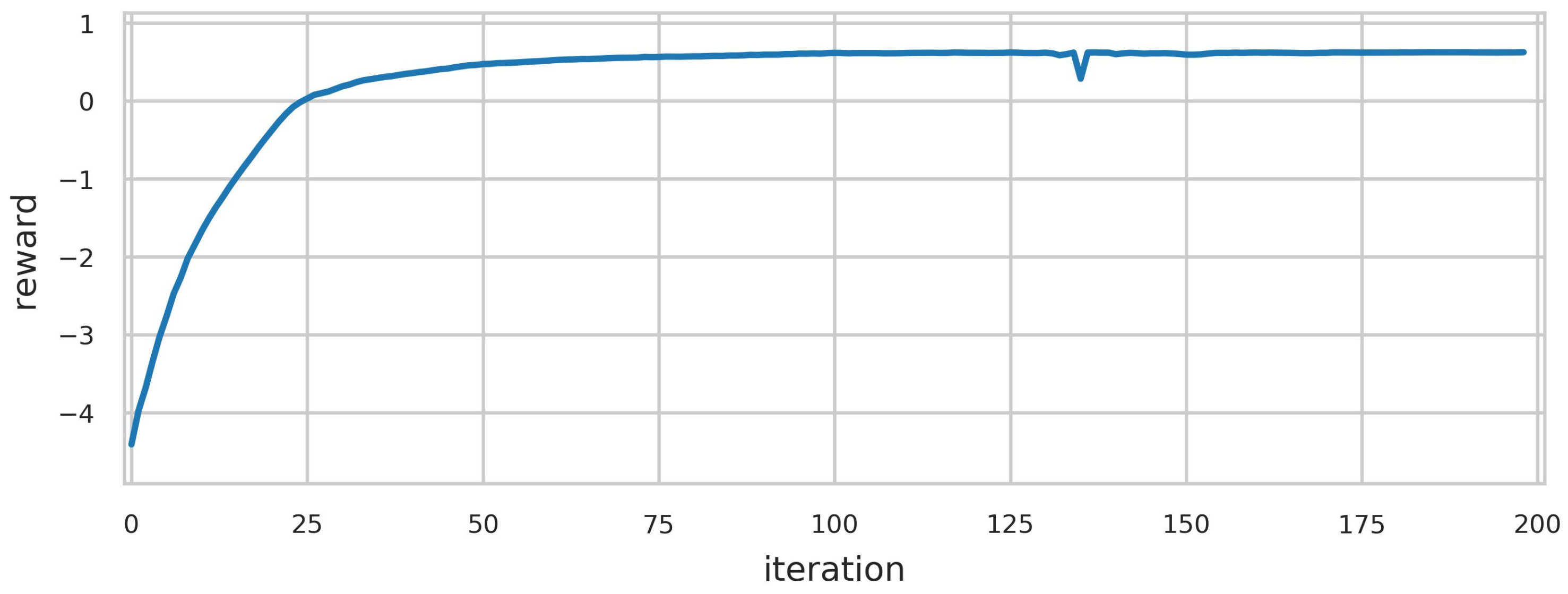
iteration 0



iteration 199



batch size:100 lr:0.005 iteration:199



Conclusions:

We used succesfully a model-free reinforcement learning to optimise a constrained continuous control problem. Furthermore, we showed that transfer learning techniques may be used to reduce the impact of model-plan mismatch.

Future improvements:

A valuable extension would be to utilize surrogate models based on real plant measurements to simulate the real plan dynamics with uncertainty and test the effectiveness of our current implementation agains this more realistic highly uncertain real plan model.

Other interesting direction is to improve the sample complexity via improved baseline estimator leveraging modern sampling without replacement techniques as stochastic beam search.

References:

- Peters, J. and Schaal, S., 2008. Reinforcement learning of motor skills with policy gradients. Neural networks, 21(4), pp.682-697.
- Chou, P.W., Maturana, D. and Scherer, S., 2017, August. Improving stochastic policy gradients in continuous control with deep reinforcement learning using the beta distribution. In Proceedings of the 34th International Conference on Machine Learning-Volume 70 (pp. 834-843). JMLR. org.
- Sutton, R.S. and Barto, A.G., 2018. Reinforcement learning: An introduction. MIT press.
- Williams, R.J., 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning, 8(3-4), pp.229-256.
- Rumelhart, D.E., Hinton, G.E. and Williams, R.J., 1988. Learning representations by back-propagating errors. Cognitive modeling, 5(3), p.1.
- Wierstra, D., Förster, A., Peters, J. and Schmidhuber, J., 2010. Recurrent policy gradients. Logic Journal of the IGPL, 18(5), pp.620-634.

[a] School of Chemical Engineering and Analytical Science,The University of Manchester, M13 9PL, UK
[b] No affiliation
[c] Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway