Forward Models in the Cerebellum using Reservoirs and Perturbation Learning

TECHNISCHE UNIVERSITÄT CHEMNITZ

Katharina Schmid, Julien Vitay and Fred Hamker

Artificial Intelligence, Department of Computer Science, TU Chemnitz.

julien.vitay@informatik.tu-chemnitz.de https://tu-chemnitz.de/informatik/KI

Abstract

The cerebellum is thought to be able to learn forward models, which allow to predict the sensory consequences of planned movements and adapt behavior accordingly. Although classically considered as a feedforward structure learning in a supervised manner, recent proposals highlighted the importance of the internal recurrent connectivity of the cerebellum to produce rich dynamics (Rössert et al., 2015), as well as the importance of reinforcement-like mechanisms for its plasticity (Bouvier et al., 2018). Based on these models, we propose a neuro-computational model of the cerebellum using an inhibitory reservoir architecture and biologically plausible learning mechanisms based on perturbation learning. The model is trained to predict the position of a simple robotic arm after ballistic movements. Understanding how the cerebellum is able to learn forward models might allow elucidating the biological basis of model-based reinforcement learning.

Computational model

Cortical inputs reach the cerebellum through the pons, which sends mossy fibers to the 1000 **granule cells** (GC) and 100 **Golgi cells** (GoC). As formalized by Rössert et al. (2015), the GC-GoC recurrent network forms a reservoir allowing to represent complex dynamics even in the absence of stimulation. Granule cells send parallel fibers along the surface of the cerebellum, which are read out by 20 **Purkinje cells** (PC) inhibiting 2 **dentate nucleus** (DN) neurons, the output of the cerebellum. 2 **inferior olive** (IO) neurons send a binary error signal to PCs, allowing to adapt the parallel fibers-PC synapses.

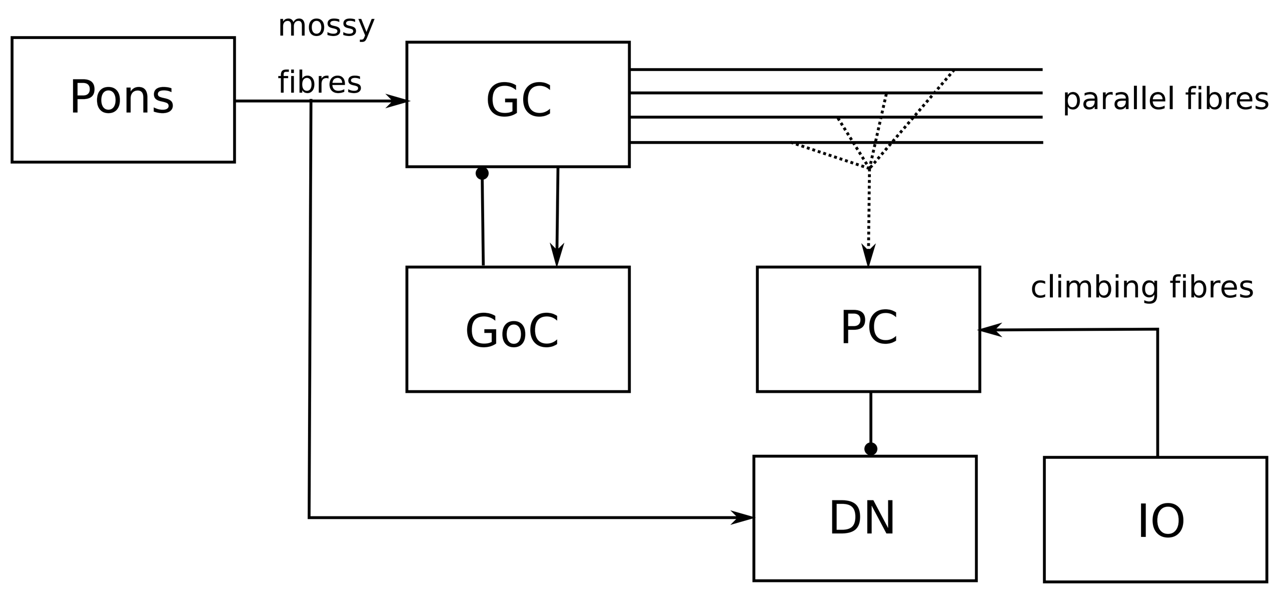

Figure 1: Network architecture: GC - granule cells, GoC -Golgi cells, PC - Purkinje cells, DN - dentate nucleus, IO - inferior olive. Edges ending with an arrow head indicate excitatory connections, edges ending in a dot indicate inhibitory connections. Dashed lines are plastic.

Granule-Golgi Network as a Reservoir

The neuron model of GC, GoC, PC and DN neurons use synaptic integration to exhibit various dynamics:

$$r_i(t) = \left[\sum_j^{N_z} w_{ij} \sum_{s=1}^{t} \exp\left(-\frac{t-s}{\tau}\right) r_j(s-1) \right]^+$$

The excitatory-inhibitory GC-GoC network exhibit rich dynamics after the presentation of a short impulse input:

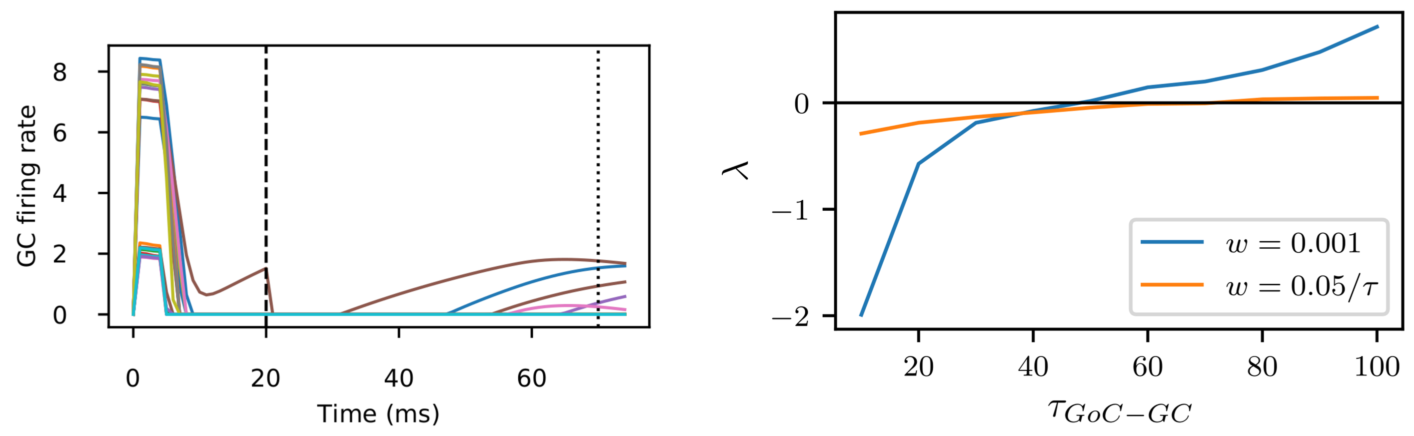

Figure 2: a) Activity of 20 selected granule cells after a short impulse input. b) Lyapunov exponent depending on synaptic time constants.

Perturbation Learning

As in Bouvier et al. (2018), each PC receives random perturbations from the IO during a trial. This perturbation modifies the PC's rate and can either improve or worsen the overall performance. Each synapse between the i-th GC (rate $z_i(t)$) and the j-th PC (rate $p_j(t)$, perturbation $I_j(t)$) maintains an eligibility trace $e_{ij}(t)$ according to:

$$e_{ij}(t) = e_{ij}(t-1) + \left(\sum_{s=1}^{t} \exp\left(-\frac{t-s}{\tau_m}\right) z_i(s-1) \right) \times \left(\sum_{s=1}^{t} \exp\left(-\frac{t-s}{\tau_{IO}}\right) I_j(s-1) \right)$$

At the end of a trial at time T, PF-PC synaptic weights are updated according to:

$$\Delta w_{ij} = -\eta \, c \, e_{ij}(T)$$

where c is a simple binary reinforcement signal indicating whether the global performance improved after the perturbation or not:

$$c = \text{sign}(\epsilon^\mu - I^\mu)$$

The current error ϵ^μ for an input μ is the Euclidean distance between the network output and the target value. The average of recent errors for a specific input pattern I^μ is estimated by a linear combination of the GC outputs $I^\mu = \sum_i v_i z_i(T)$, where $z_i(T)$ is the firing rate of the i-th GC at the end of a trial and the coefficient v_i is updated at the end of each trial according to:

$$\Delta v_i = \eta \, c z_i(T)$$

Contrary to Bouvier et al. (2018), weights are restricted to be positive.

Forward model of a 2D arm

We train the model to predict the next position x_{t+1}, y_{t+1} of the end effector of a 2D arm with two DoF θ_1 and θ_2 based on the current position x_t, y_t and the motor command $\Delta\theta_1, \Delta\theta_2$. The network is presented with the four input signals $x_t, y_t, \Delta\theta_1$ and $\Delta\theta_2$ for 20 ms. After a delay of 50 ms, the network response is read out. During this response period of 5 ms, the Purkinje cells receive random perturbations that modify the cerebellar output. Perturbations are generated randomly and independently by each IO neuron with a mean rate of 50 Hz and an amplitude of 0.1. Finally, the network response is evaluated, compared with the desired positions, and the parallel fibre-Purkinje cell weights are updated. The network is trained on a set of 5,000 random samples for 2,000 epochs and its performance is evaluated on a test set of another 5,000 random samples.

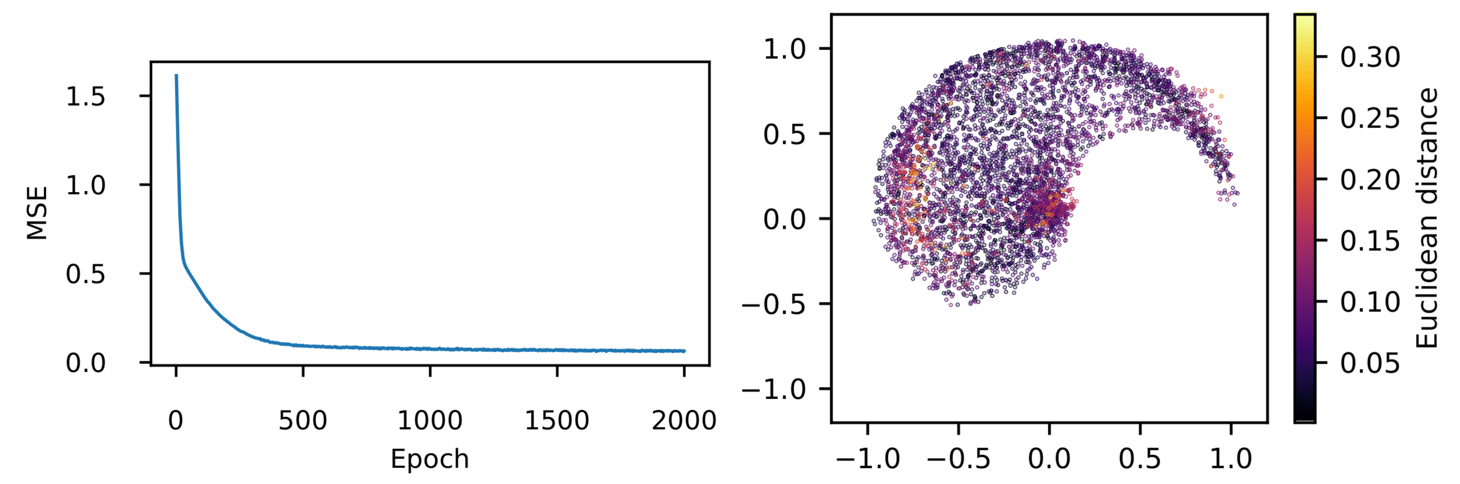

Figure 3: a) Development of the prediction error during perturbation learning with positivity constraint. b) Distribution of the prediction error in the arm's workspace.

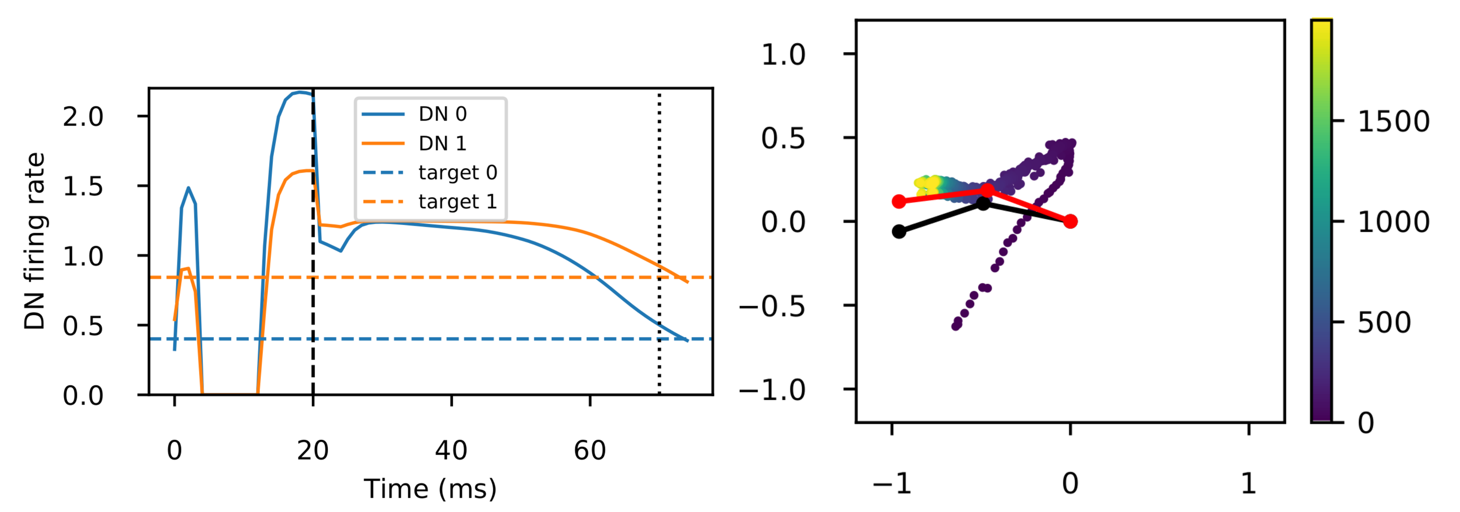

Figure 4: a) Activity of DN projection neurons for a random input sample. b) Development of the predicted arm position for a particular movement during training.

Conclusion

The proposed model combines the recurrent dynamics of the GC-GoC excitatory-inhibitory network proposed by Rössert et al. (2015) with the perturbation-based learning rule for parallel fibres-PC synapses proposed by Bouvier et al. (2018). The model is able to learn a simple non-linear prediction task on a 2D simulated arm, although still imprecisely. Contrary to the classical supervised approach requiring complete error signals, the model learns from a binary teaching signal indicating whether the prediction error has improved compared to baseline performance. This allows to learn forward models with a cerebellar model: the IO mainly receives low-level motor and proprioceptive information, so it can only drive supervised learning of inverse models (motor adaptation). For supervised forward models, the IO would need to compare cortical sensory representations in order to compute the teaching signal, what seems to be a very challenging task for such a small nucleus. By relying on a much simpler reinforcement-like teaching signal, the proposed model could learn forward models even if the predicted sensory space is high dimensional.

Code available at https://github.com/kimschmi/CerebellumForwardModel.

Acknowledgments: this work was partially supported by the grant "KIN-TUC" of the Saxonian Ministery for Science and Art (SMWK).

*

Bouvier, G., Aljadeff, J., Clopath, C., Bimbard, C., Ranft, J., Blot, A., Nadal, J.-P., Brunel, N., Hakim, V., and Barbour, B. (2018). Cerebellar learning using perturbations. *eLife*, 7.

Rössert, C., Dean, P., and Porrill, J. (2015). At the Edge of Chaos: How Cerebellar Granular Layer Network Dynamics Can Provide the Basis for Temporal Filters. *PLOS Computational Biology*, 11(10):e1004515–e1004515.