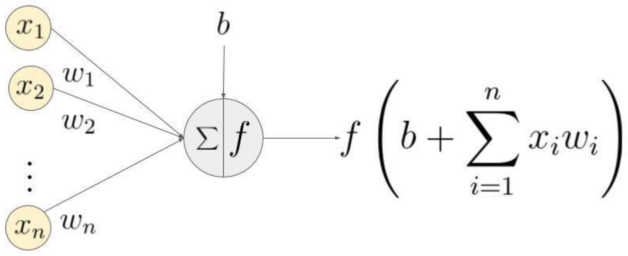
Safe Lifelong Learning: Spiking neurons as a solution to instability in plastic neural networks

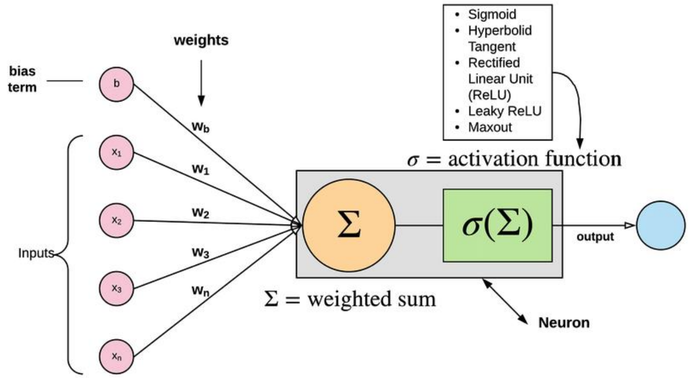
Samuel Schmidgall, Joe Hays
Robotics and Machine Learning,
Code 8234

Artificial and Spiking Neural Networks

Artificial neurons are 'stateless' and activity is produce via non-linear functions



An example of a neuron showing the input (x_1 - x_n), their corresponding weights (w_1 - w_n), a bias (b) and the activation function f applied to the weighted sum of the inputs.



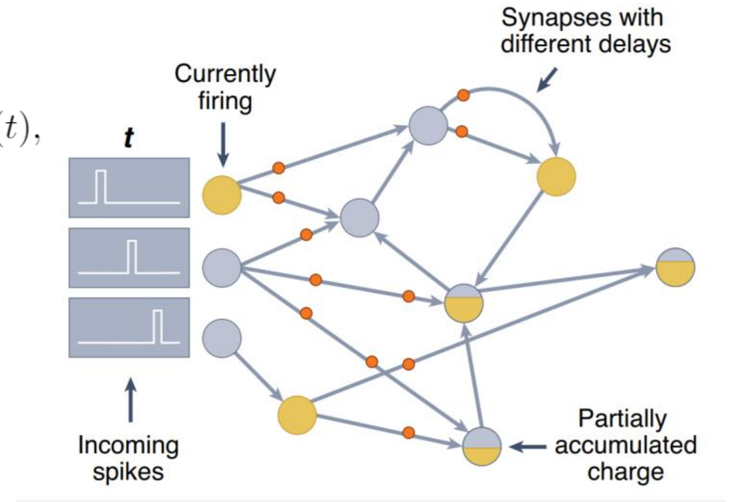
Spiking neurons accumulate information across the time domain through membrane potential, and spike when a threshold value is reached

Membrane Potential

$$\boldsymbol{v}_j(t+\Delta\tau) = \boldsymbol{v}_j(t) - \alpha_v[\boldsymbol{v}_j(t) - v_{rest}] + R\sum_i \boldsymbol{W}_{i,j}(t)\boldsymbol{s}_i(t),$$

Spiking Neuron

$$\boldsymbol{s}_j(t) = H(\boldsymbol{v}_j(t)) = \begin{cases} 0 & \boldsymbol{v}_j(t) \le v_{th} \\ 1 & \boldsymbol{v}_j(t) > v_{th} \end{cases},$$



U.S. NAVAL RESEARCH LABORATORY

Synaptic Plasticity as a means toward intra-lifetime learning

- Synaptic plasticity is thought to be one of the primary mechanisms of learning in the brain.
- Plasticity rules change synaptic weight based on local activity

ABCD Rule

- Flexible Learning Rule
- Coefficients on joint activity, pre, post and bias
- Learning rate determines magnitude and direction

Pair-based STDP

- Precise spike-timing determines weight change
- Depression if more post-without-pre
- Potentiation if more pre-before-post

$$W^{(l)}(t+\delta\tau) = W^{(l)}(t) + \alpha_w^{(l)} \odot \Delta_{ABCD}(t)$$

$$\Delta_{ABCD}(t) = (A_w^{(l)} + B_w^{(l)} + C_w^{(l)} + D_w^{(l)})(t)$$

$$A_w^{(l)}(t) = A^{(l)} \odot (x^{(l)}(t)^\intercal \times x^{(l-1)}(t))$$

$$B_w^{(l)}(t) = B^{(l)} \odot (x^{(l)}(t)^\intercal \times \mathbf{1}_{(l-1)})$$

$$C_w^{(l)}(t) = C^{(l)} \odot (\mathbf{1}_{(l)}^\intercal \times x^{(l-1)}(t))$$

$$\tau_+ \frac{dx}{dt} = -x_j + a_+(x_j) \sum_{pre} \delta(t - t_j^{pre})$$

$$\tau_- \frac{dy}{dt} = -y_j + a_-(y) \sum_{post} \delta(t - t^{post})$$

$$\Delta W_j^{(l)} = A_+(W_j)x(t) \sum \delta(t - t^n) - A_-(W_j)y(t) \sum \delta(t - t_j^f)$$

Evolutionary Strategies

Evolve the initial weights and *synaptic plasticity* parameters for a population of neural networks

$$x^{(l)}(t) = \sigma(W^{(l)}(t) \times x^{(l-1)}(t)),$$

Algorithm 1 Evolution Strategies

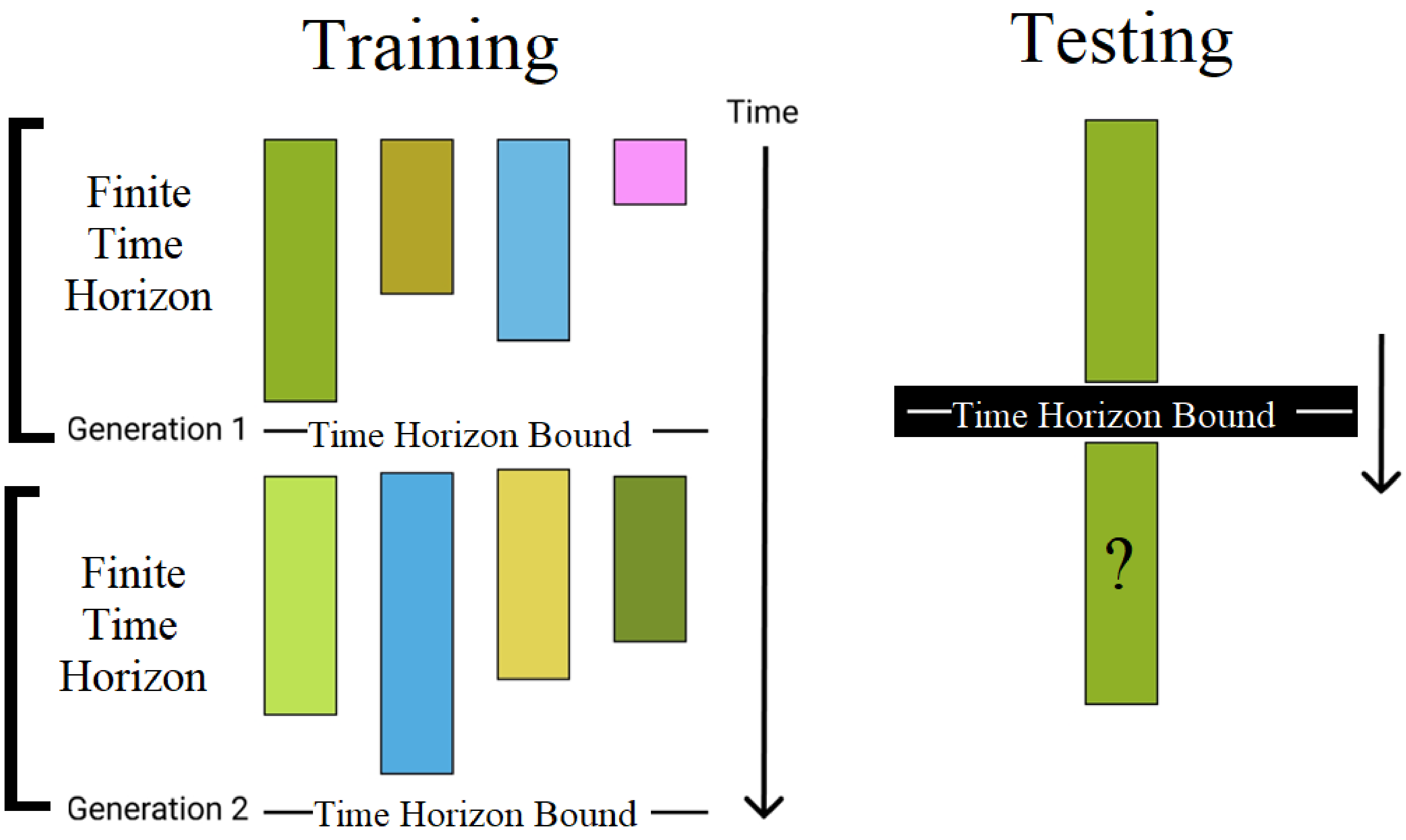
Input: Learning rate α, noise standard deviation σ, initial policy parameters θ_0
for $t = 0, 1, 2, \ldots$ **do**
  Sample $\epsilon_1, \ldots \epsilon_n \sim \mathcal{N}(0, I)$
  Compute returns $F_i = F(\theta_t + \sigma\epsilon_i)$ for $i = 1, \ldots, n$
  Set $\theta_{t+1} \leftarrow \theta_t + \alpha \frac{1}{n\sigma} \sum_{i=1}^{n} F_i \epsilon_i$
end for

$$\tau_{-} \frac{dy}{dt} = -y_j + a_{-}(y) \sum_{post} \delta(t - t^{post})$$

$$\Delta W_j^{(l)} = A_{+}(W_j)x(t) \sum \delta(t - t^n) - A_{-}(W_j)y(t) \sum \delta(t - t_j^f)$$

The problem of finite lifespan

- Time-dependent parameters are being optimized across a (short) time horizon.

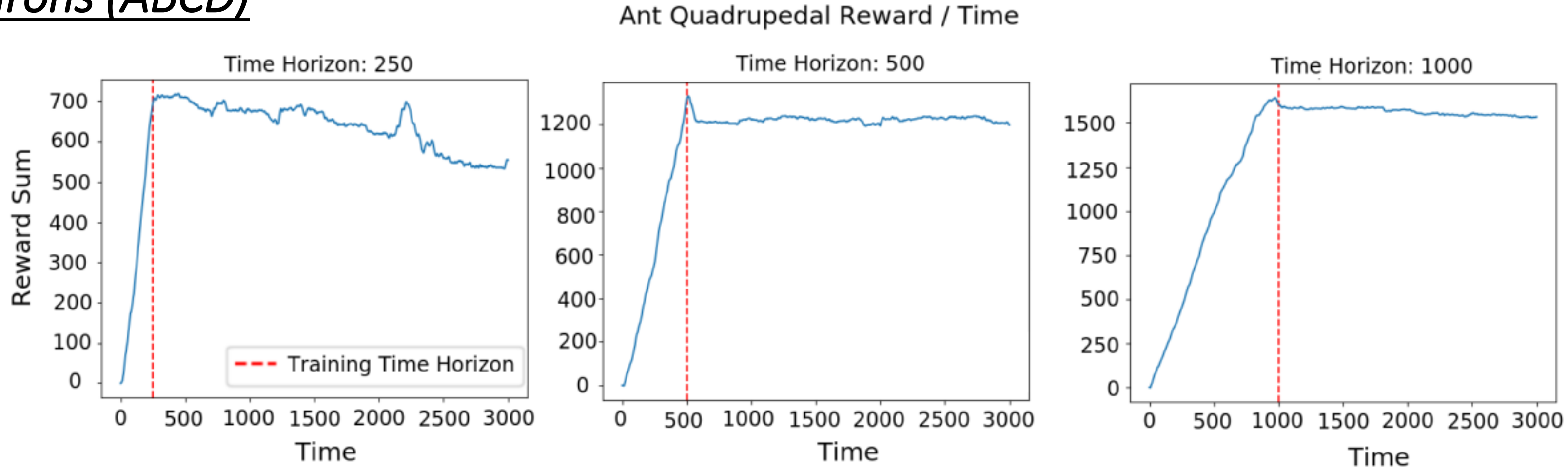


- Does intra-lifetime learning generalize to the time domain?

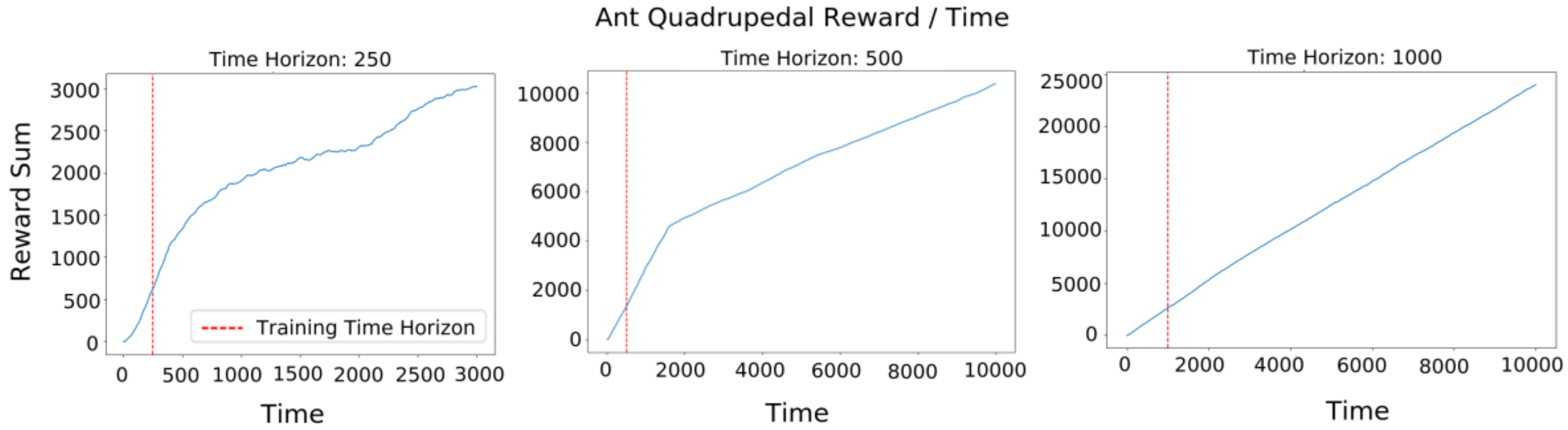
A reinforcement learning experiment

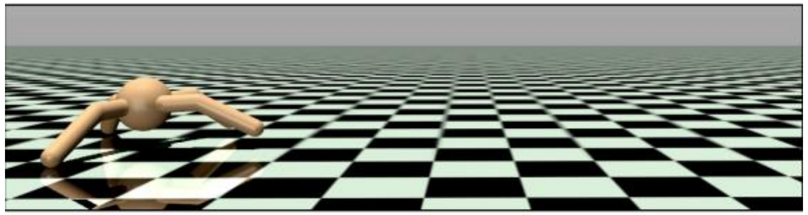
- PANNs are shown to degrade in performance instantaneously after the trained time horizon
- PSNNs are shown to continue collecting positive reward, which improved in generalization with a greater time horizon

Artificial Neurons (ABCD)



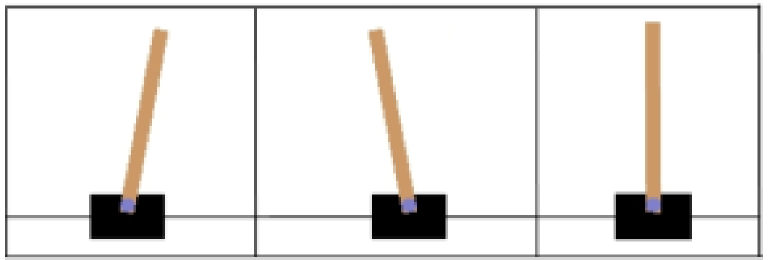
Spiking Neurons (ABCD)



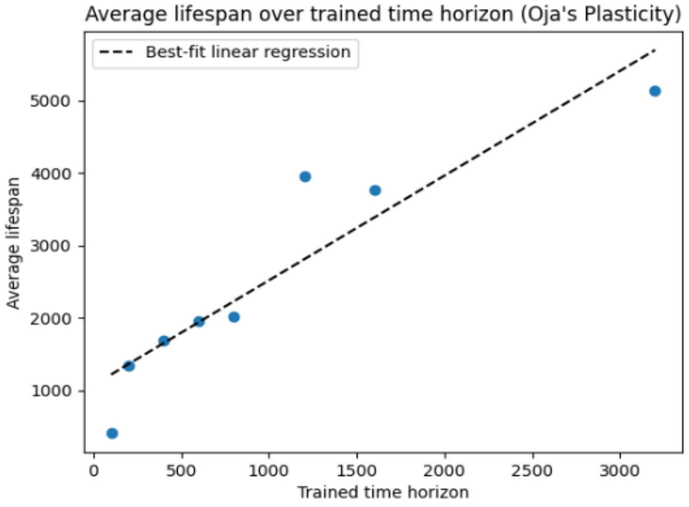


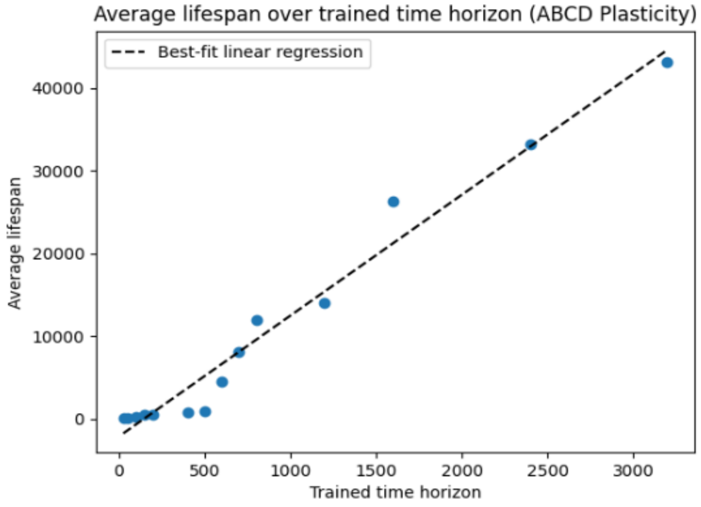
An experiment in long-term control stability

- PANNs are shown to have a linear relationship between the time-horizon and the amount of time balanced
- PSNNs are shown to be capable of balancing the pole indefinitely for any time horizon beyond 400 for all tested plasticity types and with recurrent PSNNs

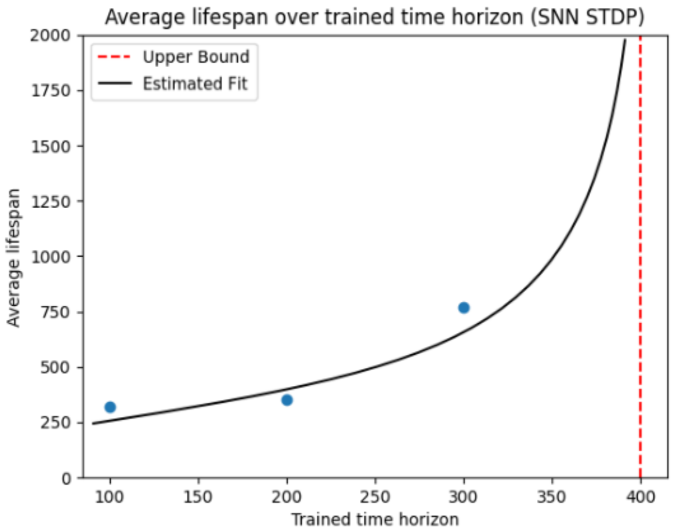


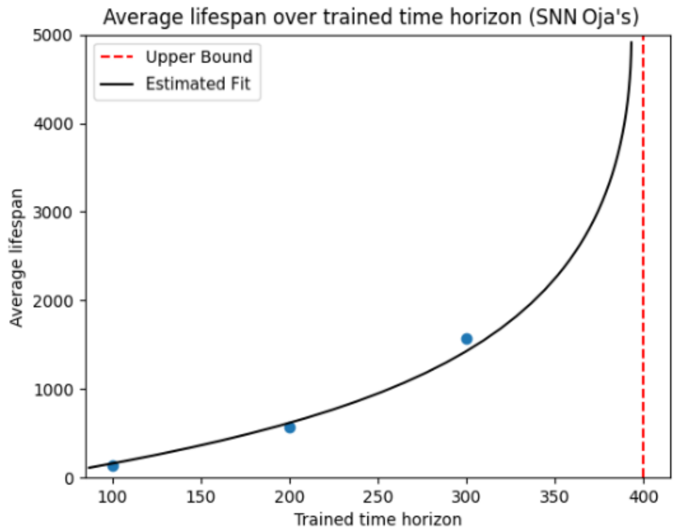
Artificial Neurons





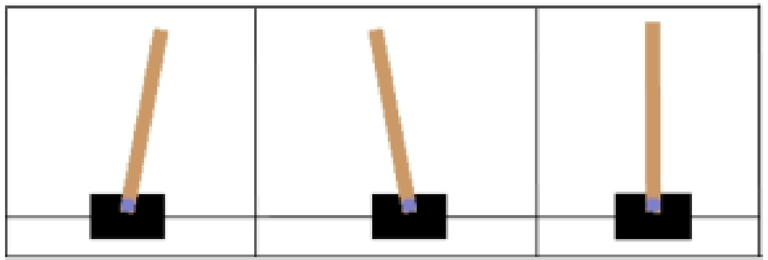
Spiking Neurons

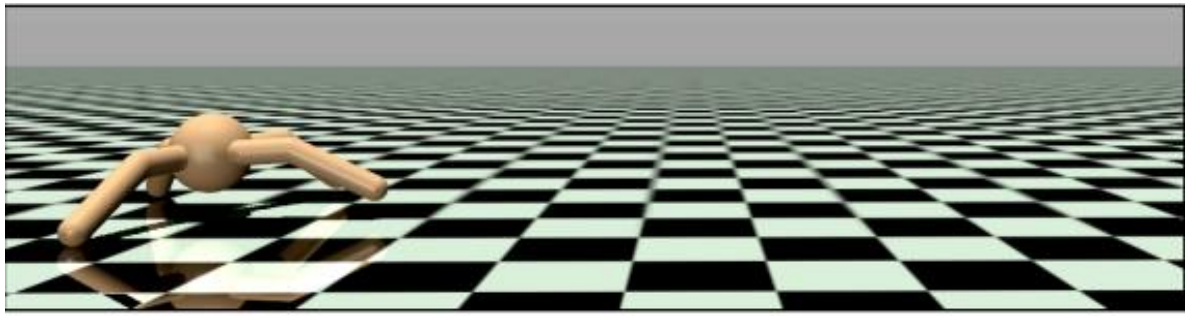


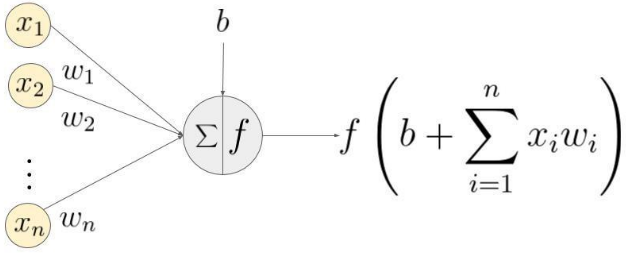


Conclusion

- The purpose of synaptic plasticity is to allow learning to occur **within and beyond the training period** of a neural network, and hence it is necessary to consider the ability to generalize not only in the task domain **but also in the time domain**
- Spiking neurons seem to generalize better in the time domain on robotic control tasks







An example of a neuron showing the input (x_1 - x_n), their corresponding weights (w_1 - w_n), a bias (b) and the activation function f applied to the weighted sum of the inputs.