

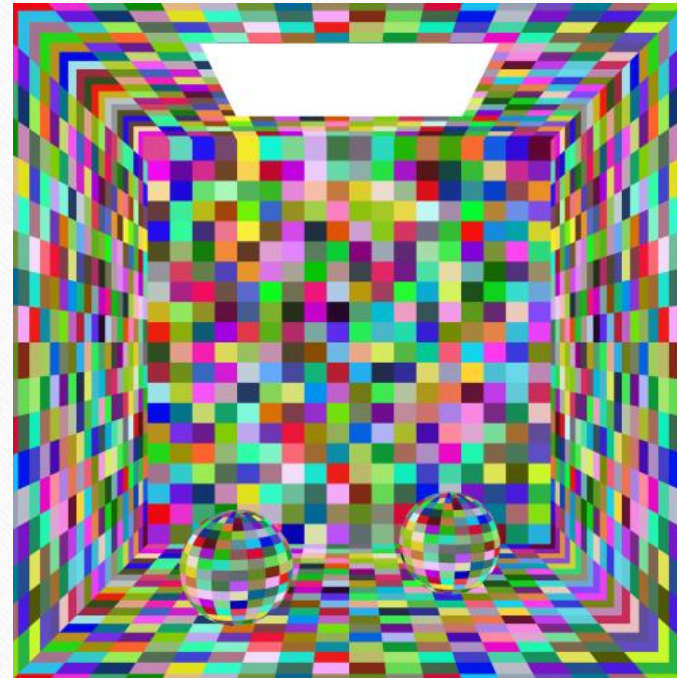
Deep Reinforcement Learning and Ray Tracing

Rendering Equation

- $L(x, \omega_0) = L_e(x, \omega_0) + \int_{H^+(x)} L_i(y, -\omega_i) f_r(x, \omega_i \rightarrow \omega_0) \cos\theta_i d\omega_i$
- $y = \text{hitpoint}(x, -\omega_i)$

Reinforcement Learning

- States: points on surfaces
- Actions: directions
- Q-learning :
Discretize states and actions
- Deep Q-learning :
Only discretize actions



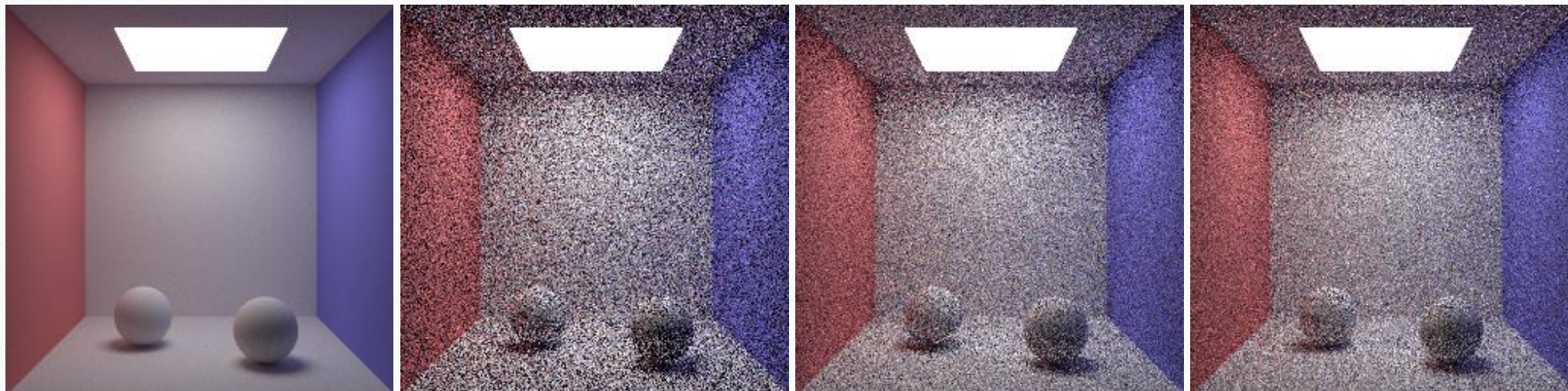
Deep Reinforcement Learning

- $Q(s, a) \sim L(x, \omega)$
- N patches to sample directions
- $target(s') \cong \frac{2\pi}{n} \sum_{i=1}^N Q(s', a'(\xi_i)) f_r \cos\theta(\xi_i)$
- $Q(s, a, w)$ is the output of a neural network called Q-network
- Loss function : $E_{(s,a,s',r) \in U(D)} (Q(s, a, w) - target(s', w^-))^2$
- Soft update of target parameters : $w^- = 0.999 \times w^- + 0.001 \times w$

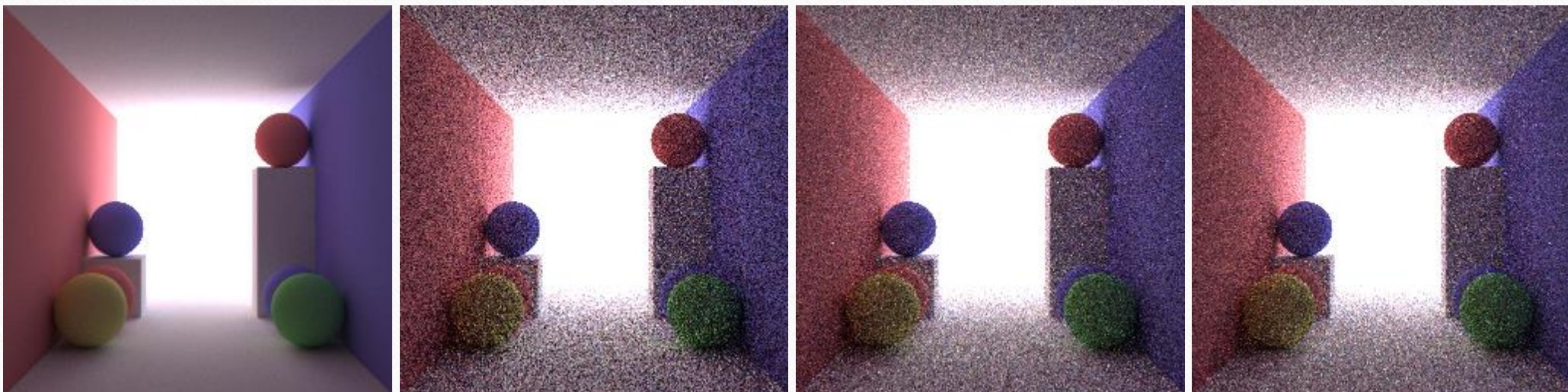
MLP Inputs

- Two different types of inputs
- $input1 = (x, n)$
- $input2 = (x, \gamma(x), n)$
- n : normal of point
- γ : positional encoding
- $\gamma(x) = (\sin(2^0\pi x), \cos(2^0\pi x), \dots, \sin(2^{L-1}\pi x), \cos(2^{L-1}\pi x))$

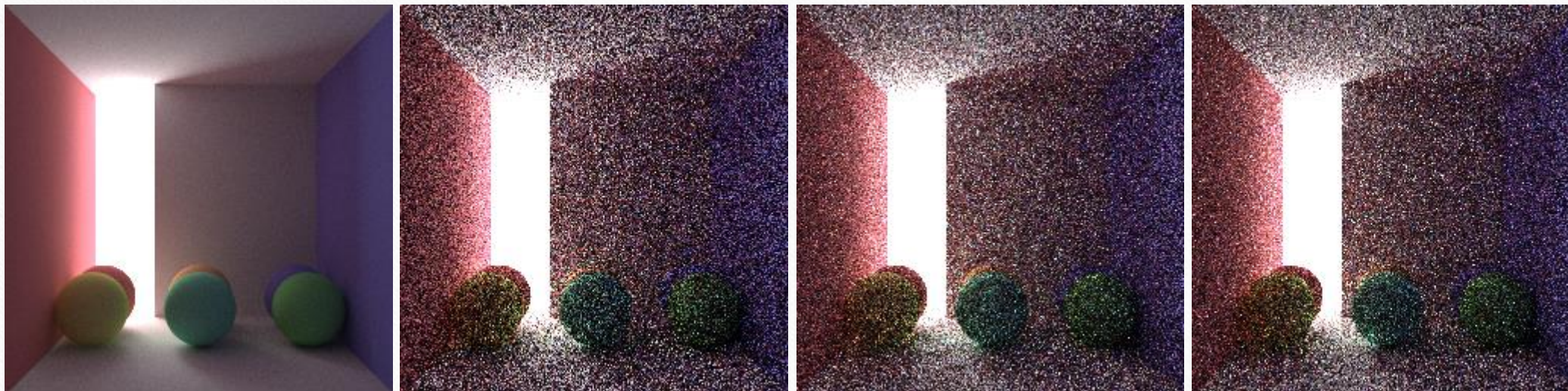
Scene 1: Uniform – DQN – DQN-POS



Scene 1: Uniform – DQN – DQN-POS

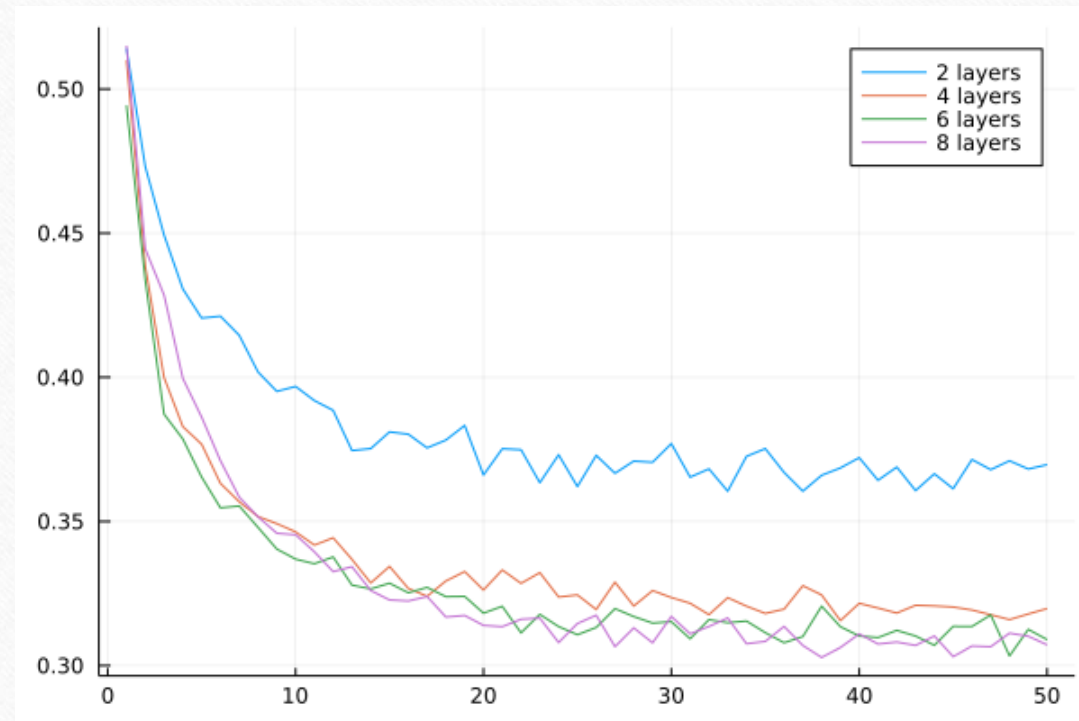


Scene 1: Uniform – DQN – DQN-POS



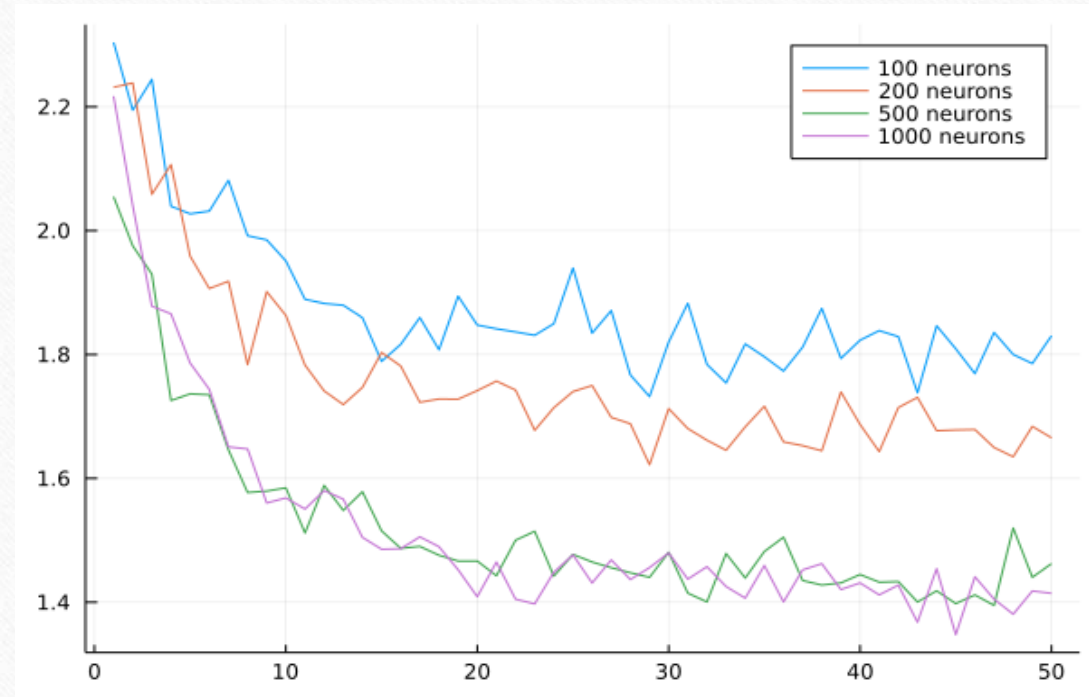
Loss Function : Depth

- Number of layers: {2,4,6,8}
- {Dense(54, H, relu), ..., Dense(H, 108, softplus)}
- Each layers has 100 neurons



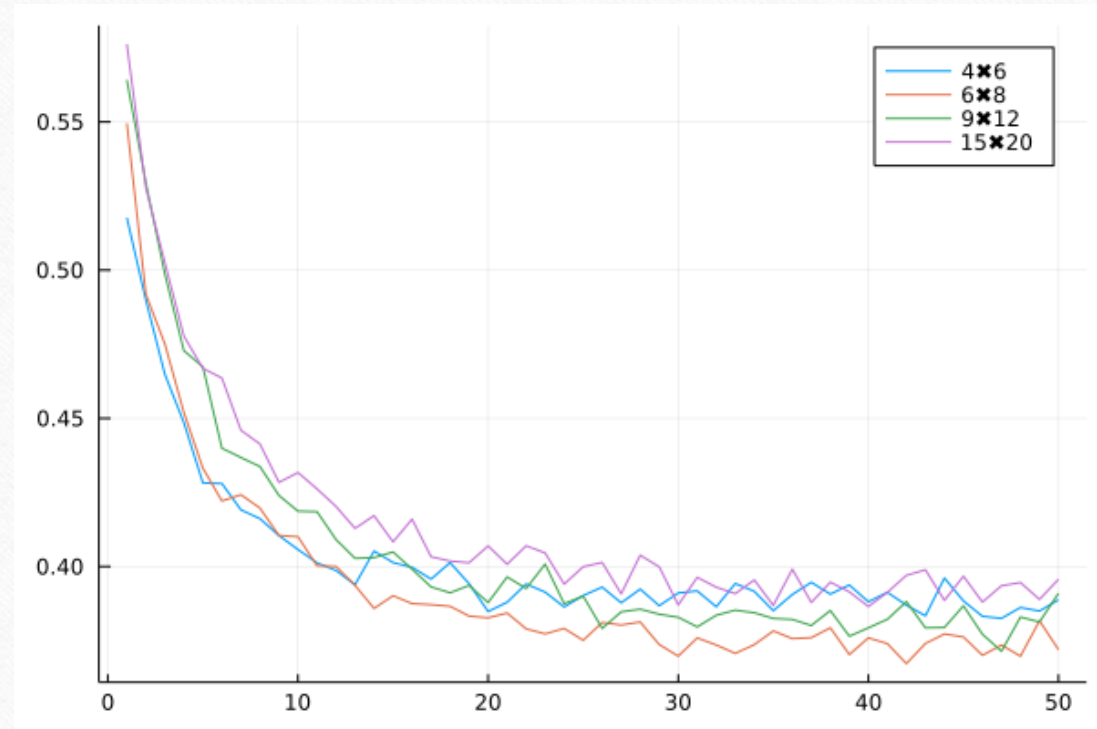
Loss Function : Width

- Two-layer neural networks:
- {Dense(54, H, relu),
Dense(H, 108, softplus)}
- H values plot



Loss Function : Discretization Resolution

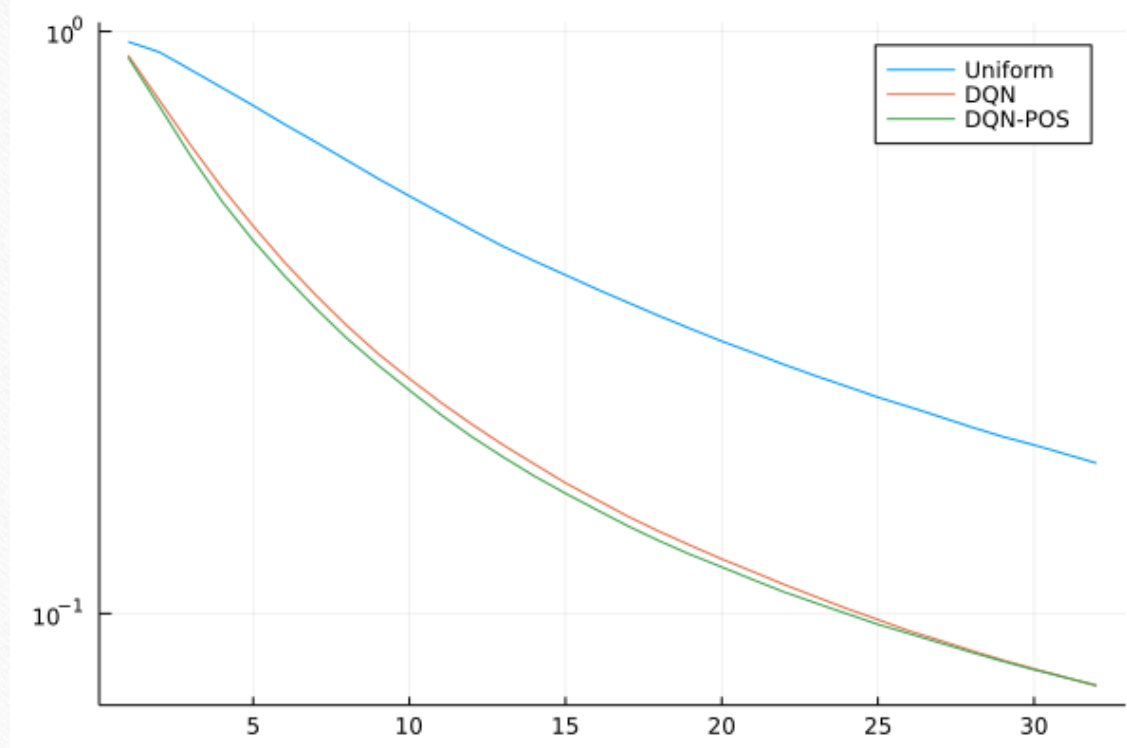
- Action space resolutions
 $\{4 \times 6, 6 \times 8, 9 \times 12, 15 \times 20\}$



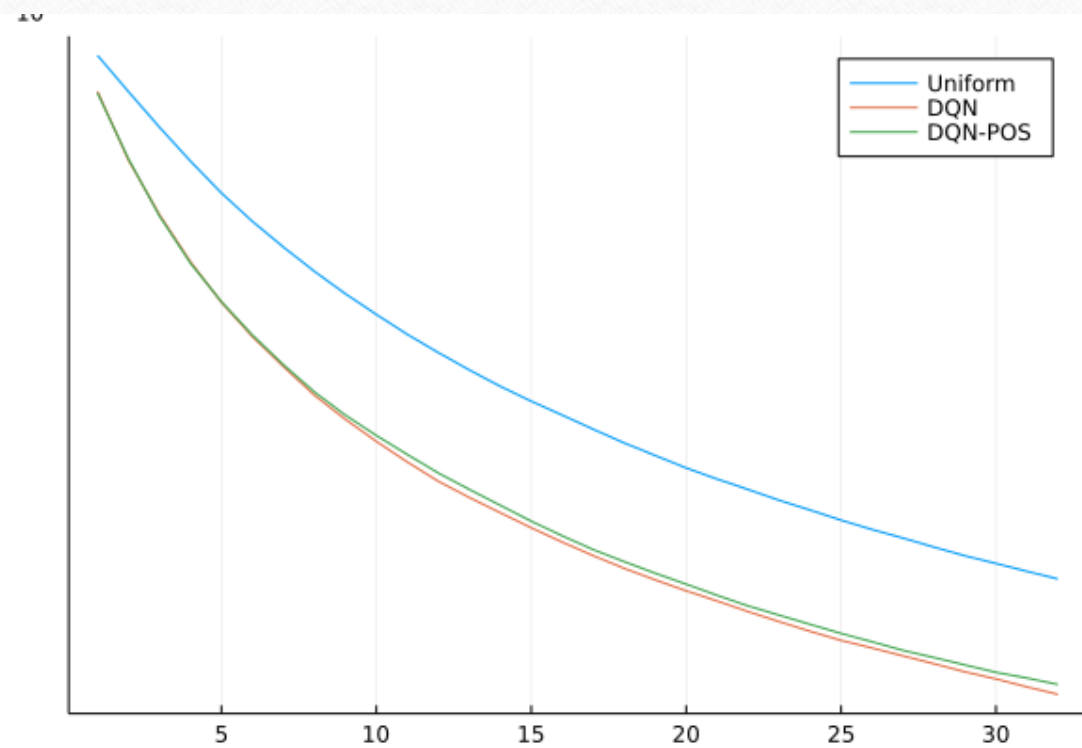
Network

- Two-layer neural networks
- {Dense(in, 1000, relu),
Dense(1000, 108, softplus)}
- In = {6, 54}

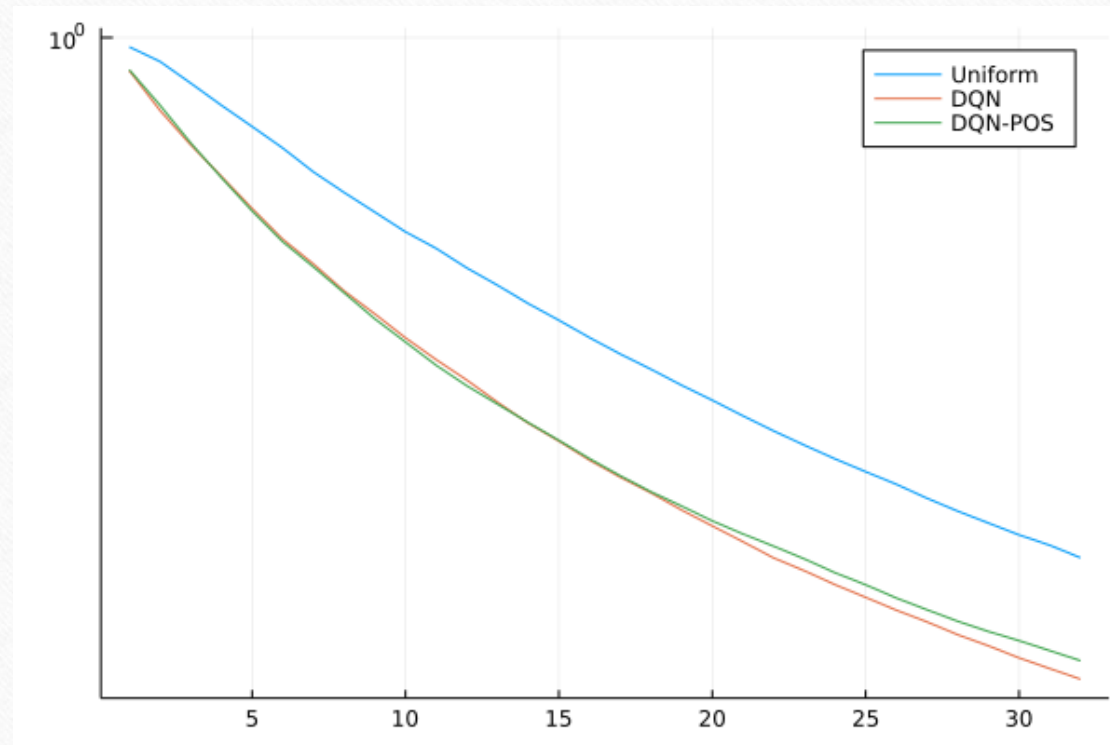
MAE



MAE



MAE



Deep Convergent Q-Learning

- C-DQN algorithm: (T. Wang)
- $L_{C-DQN}(\theta, \bar{\theta}) = \mathbb{E} [\max\{l_{DQN}(\theta, \bar{\theta}), l_{MSBE}(\theta)\}]$
- $L_{MSBE}(\theta) = L_{DQN}(\theta, \theta)$

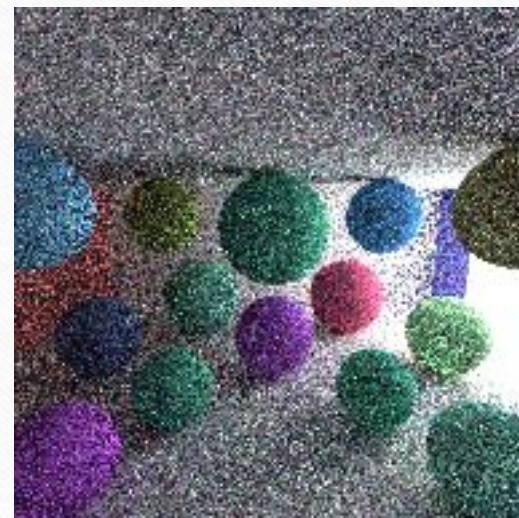
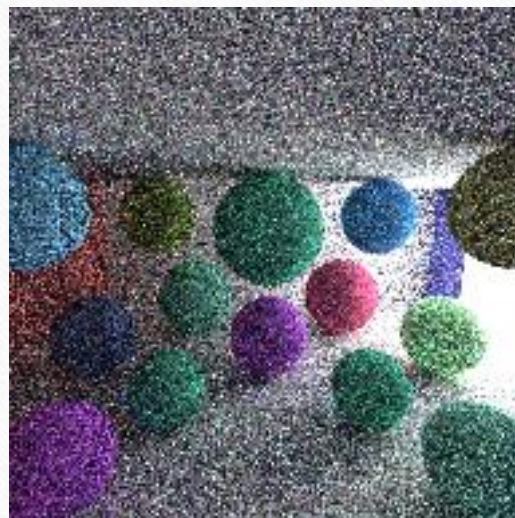
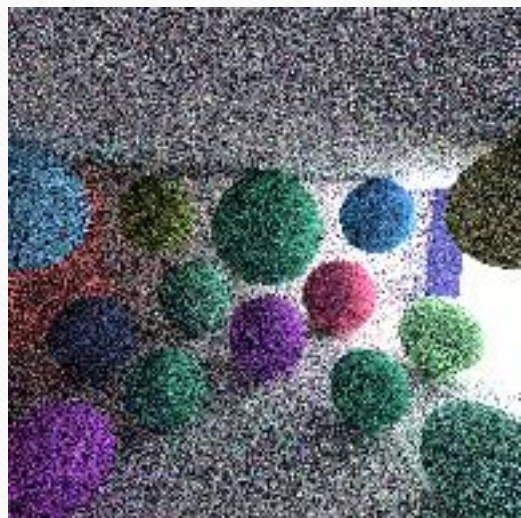
Deep RBF Network

- $RBF_{\theta}(\mathbf{x}, \mathbf{c}) = NN_{\theta}(\phi_{\mathbf{c}}(\mathbf{x}))$
- $\phi_{\mathbf{c}}(\mathbf{x}) = \exp(-0.5 \|\mathbf{x} - \mathbf{c}\|^2)$
- Divide space into $6*3*6$ cubes and consider their centers.
- $MLP = \{\text{Chain}(\text{Dense}(6*3*6, 64, \text{relu}), \text{Dense}(64, 64, \text{relu}), \text{Dense}(36, 24))\}$

Results: Uniform , C-DQN , C-DQN-RBF



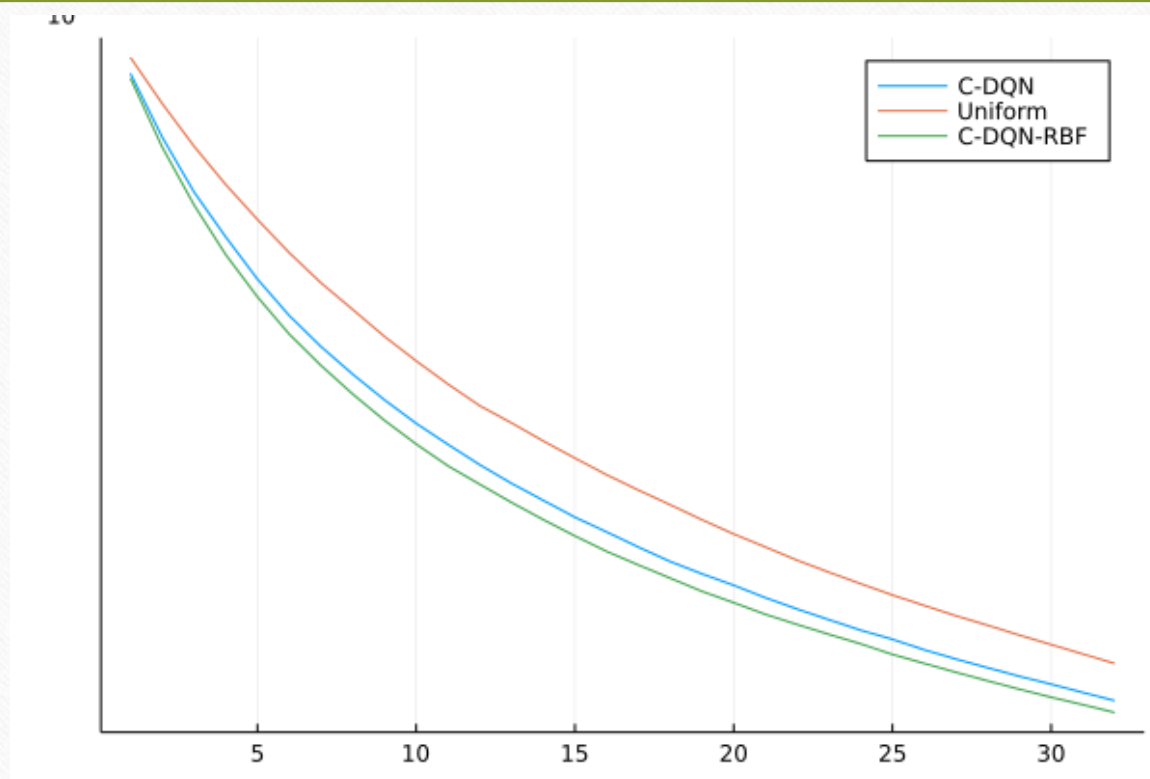
Results: Uniform , C-DQN , C-DQN-RBF



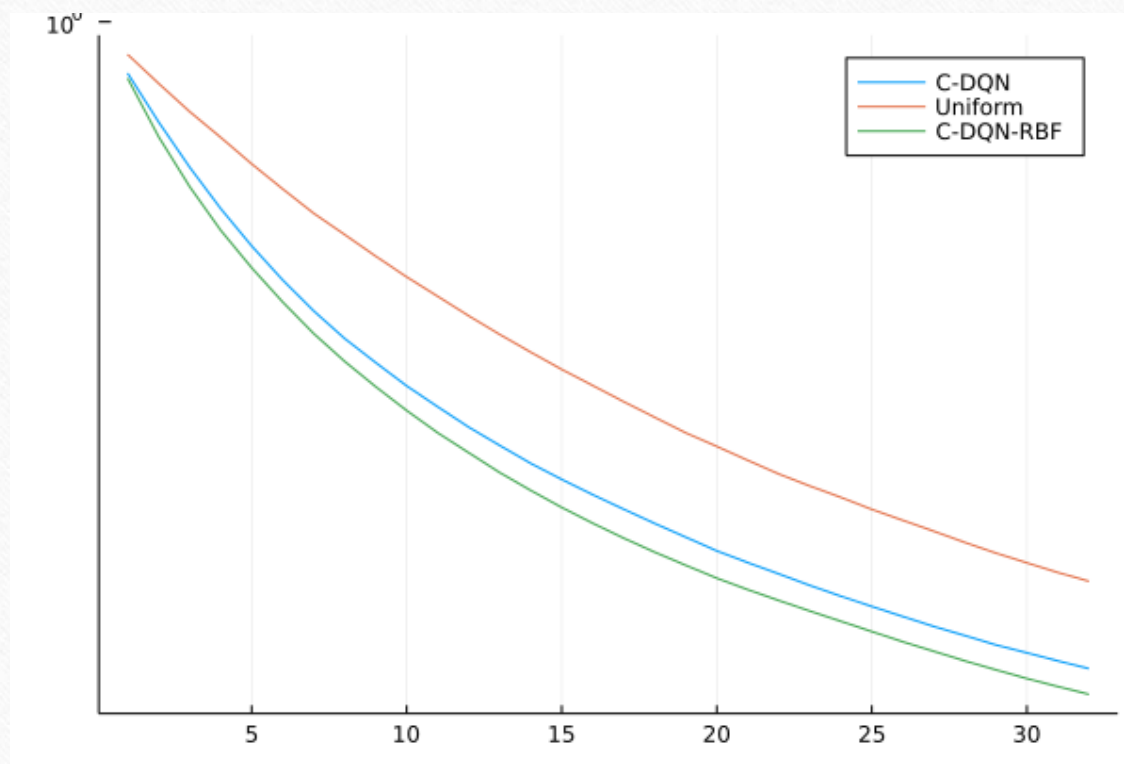
Results: Uniform , C-DQN , C-DQN-RBF



MAE



MAE



MAE

