BMVC 2022 # 358

Biologically Plausible Variational Policy Gradient with Spiking Recurrent Winner-Take-All Networks

Zhile Yang[1], Shangqi Guo[†2], Ying Fang[3], Jian K. Liu[†1]
[1]University of Leeds, [2]Tsinghua University, [3]Fujian Normal University

Keywords: reinforcement learning, spiking neural networks

Message: + SVPG, a spiking-based variational policy gradient method with RWTA network and R-STDP.
+ Experiment results reveal its potential for solving RL tasks and to have inherent robustness.

Code

1. Background & Problem

Background

RL
- ANN —Human-level?→ Adaptability, robustness ...
- SNN

SNN	Computational Efficiency	Biological Plausibility (training)	Performance
ANN-to-SNN conversion	✘	✘	✓
Surrogate gradient-based BP	✓	✘	✓
Modulated STDP	✓	✓	✘
...			

This paper

Rely on expert designs

Problem

LIF neuron model $\rho(l)=\exp\{u(l)-u_{\text{th}}\}$ $\quad u(l)=\sum_{j\in N(\cdot)} w_j\int_0^\infty \kappa(y)S_j(l-y)\mathrm{d}y+b$

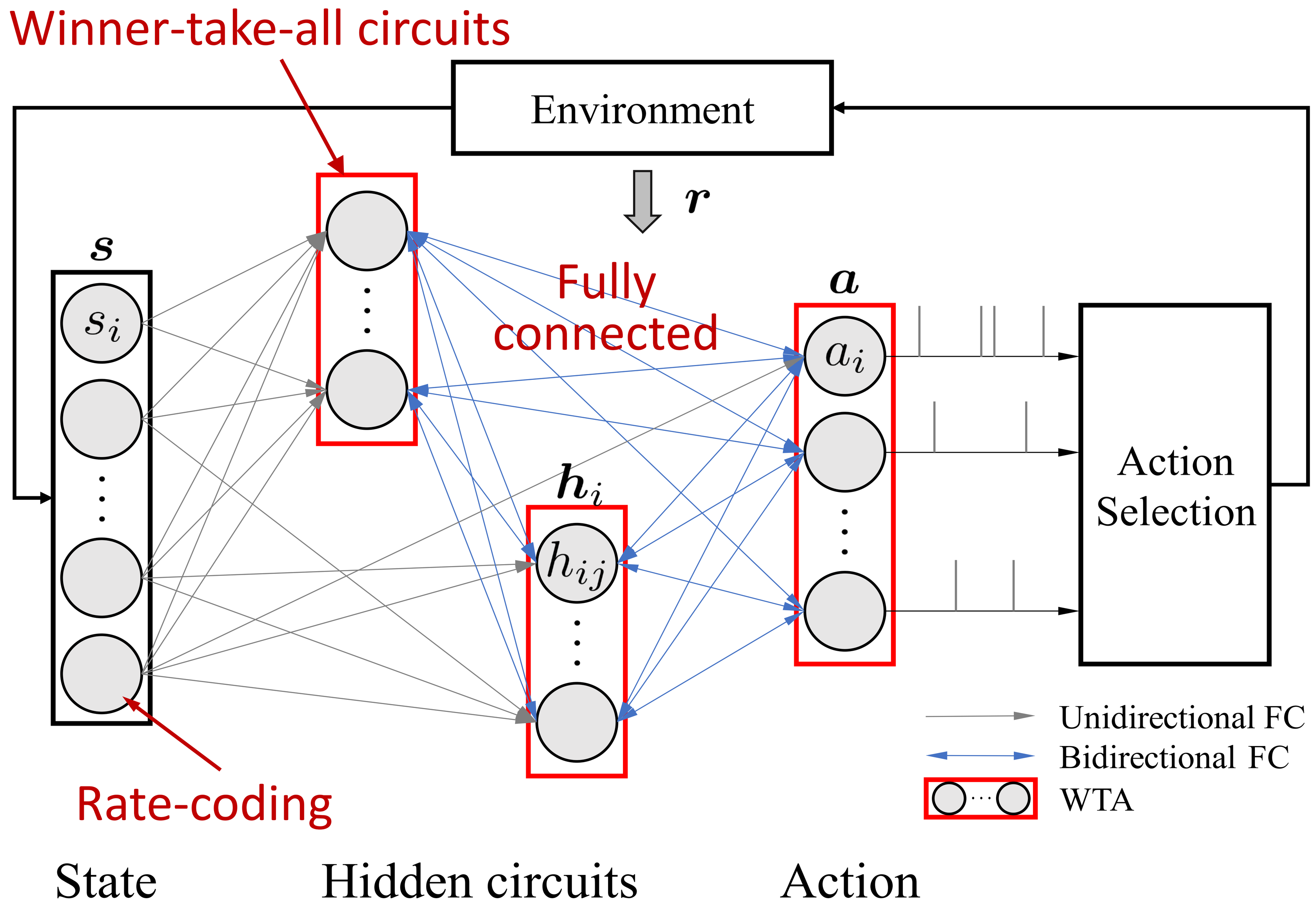
Local learning rules
$$\begin{aligned}\mathrm{STDP}(l)=&S_j(l)\left[W_{\text{pre}}+\int_0^\infty A_+W_+(y)S_i(l-y)\mathrm{d}y\right]\\&+S_i(l)\left[W_{\text{post}}+\int_0^\infty A_-W_-(y)S_j(l-y)\mathrm{d}y\right]\end{aligned}$$

How to establish the connection?

Global RL target $\nabla_\theta J(\pi_\theta)=\mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T-1}\gamma^t r_t\sum_{k=0}^{T-1}\nabla_\theta\log\pi_\theta(a_k|s_k)\right]$

2. Designs

RWTA Network



Policy Function

Policy π based on an energy function of the firing states.

$\pi(\boldsymbol{v}_a|s)=\sum_{\boldsymbol{v}_h}p(\boldsymbol{v}_a,\boldsymbol{v}_h|s)$ $\quad p(\boldsymbol{v}_a,\boldsymbol{v}_h|s):=\frac{1}{Z(s)}\exp\{E(\boldsymbol{v})\}$ $\quad E(\boldsymbol{v}):=\boldsymbol{v}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{v}+\boldsymbol{b}^{\mathrm{T}}\boldsymbol{v}$

$\boldsymbol{v}$: firing states of the neurons; $\boldsymbol{q}$: firing probabilities of the neurons

3. Policy Inference & Optimization

Policy Inference

Target function $p(\boldsymbol{v}_a,\boldsymbol{v}_h|s):=\frac{1}{Z(s)}\exp\{E(\boldsymbol{v})\}$

Mean-field inference $\hat{p}(\boldsymbol{v}_a,\boldsymbol{v}_h|s):=\hat{p}(\boldsymbol{v}_a|s)\hat{p}(\boldsymbol{v}_{h_1}|s)\cdots\hat{p}(\boldsymbol{v}_{h_{n_h}}|s)$

Minimize KL-divergence $\partial D_{\text{KL}}(s)/\partial q_i=0$ ➡ $q_i=\frac{1}{Z(q_{G(i)})}\exp\{\boldsymbol{w}_{\text{row},i}^{\mathrm{T}}\boldsymbol{q}+\boldsymbol{w}_{\text{col},i}^{\mathrm{T}}\boldsymbol{q}+b_i\}$

In RWTA network

Let $\rho_i(l)=\hat{\rho}q_i$, then $\boxed{\rho_i=\hat{\rho}\exp\{\boldsymbol{w}_{\text{row},i}^{\mathrm{T}}\boldsymbol{q}+\boldsymbol{w}_{\text{col},i}^{\mathrm{T}}\boldsymbol{q}+b_i-\log\sum_{j\in G(i)}\exp\{\boldsymbol{w}_{\text{row},j}^{\mathrm{T}}\boldsymbol{q}+\boldsymbol{w}_{\text{col},j}^{\mathrm{T}}\boldsymbol{q}+b_j\}\}}$

→ RWTA net is suitable for policy inference

In neuron model

Let $\int_0^\infty\kappa(y)\mathrm{d}y=1/\hat{\rho}$, then ➡ $\boxed{\begin{aligned}&\rho_i(l)=\hat{\rho}\exp\{u_i(l)-\log\sum_{j\in G(i)}\exp(u_j(l))\},\\&u_i(l)=\sum_{j\in N(i)}w_{ij}\int_0^\infty\kappa(y)S_{ij}(l-y)\mathrm{d}y+b_i.\end{aligned}}$

Policy Optimization

Policy function $q_i=\frac{1}{Z(q_{G(i)})}\exp\{\boldsymbol{w}_{\text{row},i}^{\mathrm{T}}\boldsymbol{q}+\boldsymbol{w}_{\text{col},i}^{\mathrm{T}}\boldsymbol{q}+b_i\}$ (iterate until convergence)

Precise differential ✘ Requires pseudo-inverse of $\boldsymbol{M}(\boldsymbol{W}+\boldsymbol{W}^{\mathrm{T}})$-- computationally infeasible

$$\frac{\partial\boldsymbol{q}_{ha}}{\partial w_{jk}}=\boldsymbol{M}(\boldsymbol{U}_{jk}+\boldsymbol{U}_{kj})\boldsymbol{q}+\boldsymbol{M}(\boldsymbol{W}+\boldsymbol{W}^{\mathrm{T}})\frac{\partial\boldsymbol{q}}{\partial w_{jk}},\qquad\frac{\partial\boldsymbol{q}_{ha}}{\partial b_j}=\boldsymbol{M}\boldsymbol{b}+\boldsymbol{M}(\boldsymbol{W}+\boldsymbol{W}^{\mathrm{T}})\frac{\partial\boldsymbol{q}}{\partial b_j},$$
$$\boldsymbol{M}=\mathrm{diag}(\boldsymbol{q}_{ha})[-\boldsymbol{G}_{ha}\mathrm{diag}(\boldsymbol{q})+\boldsymbol{D}_{\text{sel}}]$$

Approximation treat $\boldsymbol{q}$ on the right side as constants

$$\frac{\partial\log(q_i)}{\partial\boldsymbol{W}}=(\boldsymbol{U}_{i:}\mathrm{diag}(\boldsymbol{q})+\mathrm{diag}(\boldsymbol{q})\boldsymbol{U}_{:i})-\mathrm{diag}(\boldsymbol{q})(\boldsymbol{U}_{G(i):}+\boldsymbol{U}_{:G(i)})\mathrm{diag}(\boldsymbol{q}),$$
$$\frac{\partial\log(q_i)}{\partial\boldsymbol{b}}=\boldsymbol{u}_i-\mathrm{diag}(\boldsymbol{q})\boldsymbol{u}_{G(i)}$$

$$\boxed{\frac{\partial J(\pi)}{\partial w_{ij}}=\sum_t\gamma^t r_t[q_i(v_j-q_j)+q_j(v_i-q_i)]}$$

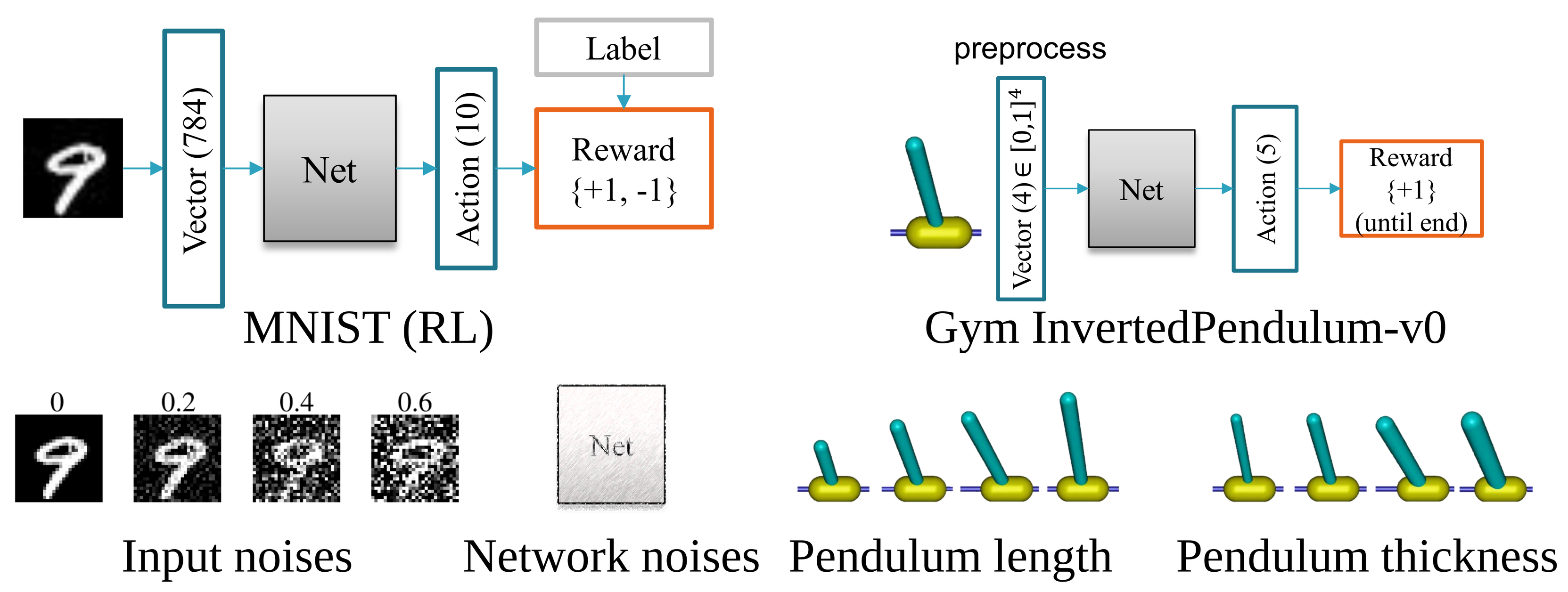
In STDP framework $\boxed{\mathbb{E}[R(l)\mathrm{STDP}(l)]=R[\rho_j(v_i-\rho_i/\hat{\rho})+\rho_i(v_j-\rho_j/\hat{\rho})]}$

1. Assume a large enough spiking step number
$\mathbb{E}[S_i(l)]=\rho_i,\quad\mathbb{E}[\int_0^\infty A_+(w_{ij})S_i(l-y)\mathrm{d}y]=A_+(w_{ij})\rho_i$
2. Let $\langle W_{\text{pre}},W_{\text{post}},A_+(w_{ij}),A_-(w_{ij})\rangle=\langle v_i,v_j,-1/\hat{\rho},-1/\hat{\rho}\rangle$

→ The designed STDP rules can do approximated policy optimization

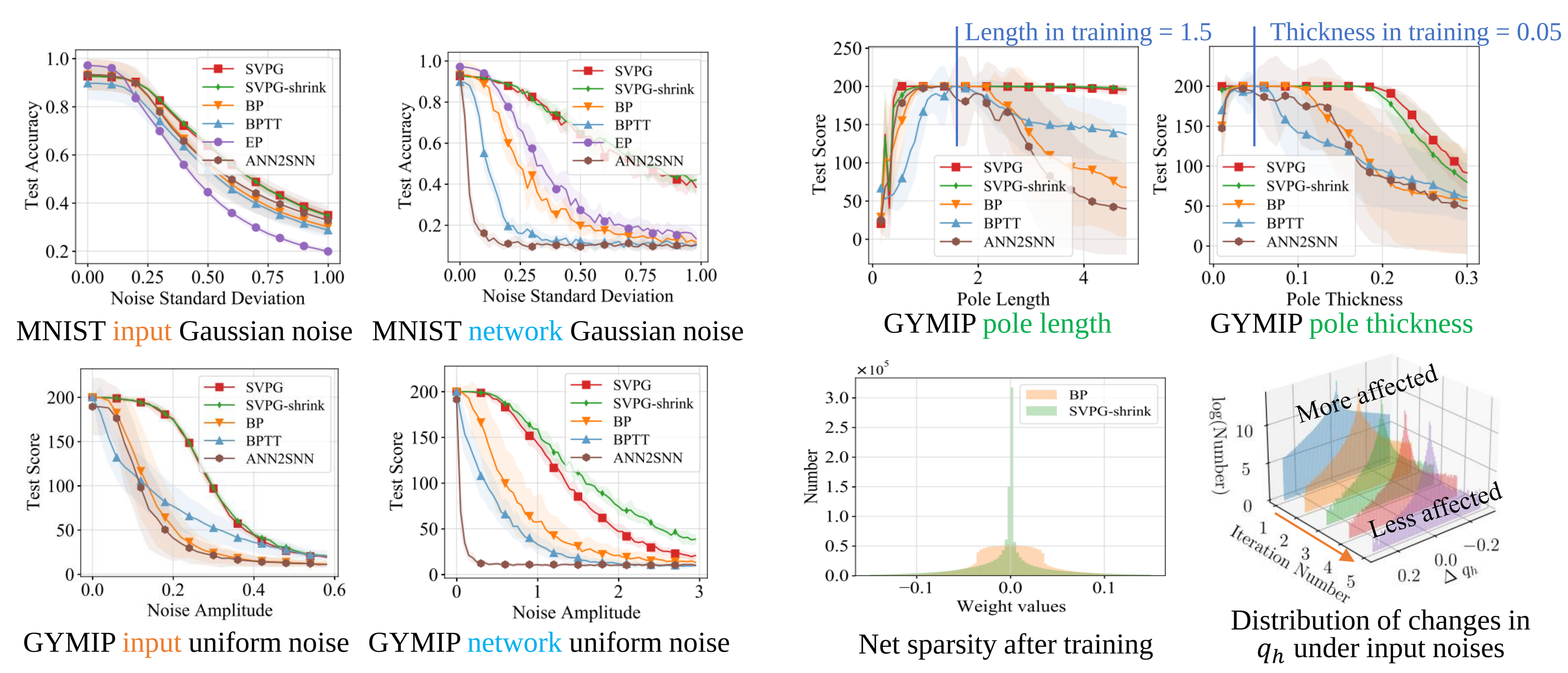
4. Experiments

Tasks & Variations



Results

	SVPG	BP	BPTT	EP(supervised)	ANN2SNN
MNIST	0.926± 0.001	0.933± 0.062	0.898± 0.067	0.971± 0.001	0.933± 0.062
GymIP	199.87 ± 0.27	199.95 ± 0.13	199.96 ± 0.12	N/A	190.79 ± 27.64



MNIST input Gaussian noise; MNIST network Gaussian noise; GYMIP pole length; GYMIP pole thickness; GYMIP input uniform noise; GYMIP network uniform noise; Net sparsity after training; Distribution of changes in q_h under input noises