電機資訊學院 2021 作 BRAIN PLUS HAND 實作專題競賽

A Distributional Perspective on Value Function Factorization Methods for Multi-Agent Reinforcement Learning (EECS09)

Wei-Fang Sun（孫偉芳）

Abstract

In fully cooperative multi-agent reinforcement learning (MARL) settings, the environments are highly stochastic due to the partial observability of each agent and the continuously changing policies of the other agents. To address the above issues, we integrate distributional RL and value function factorization methods by proposing a **Distributional Value Function Factorization (DFAC) framework** to generalize expected value function factorization methods to their distributional variants. DFAC extends the individual utility functions from deterministic variables to random variables, and models the quantile function of the total return as a quantile mixture. To validate DFAC, we demonstrate its ability to factorize a simple two-step matrix game with stochastic rewards and perform experiments on all Super Hard tasks of **StarCraft Multi-Agent Challenge** (SMAC), showing that DFAC is able to outperform expected value function factorization baselines.

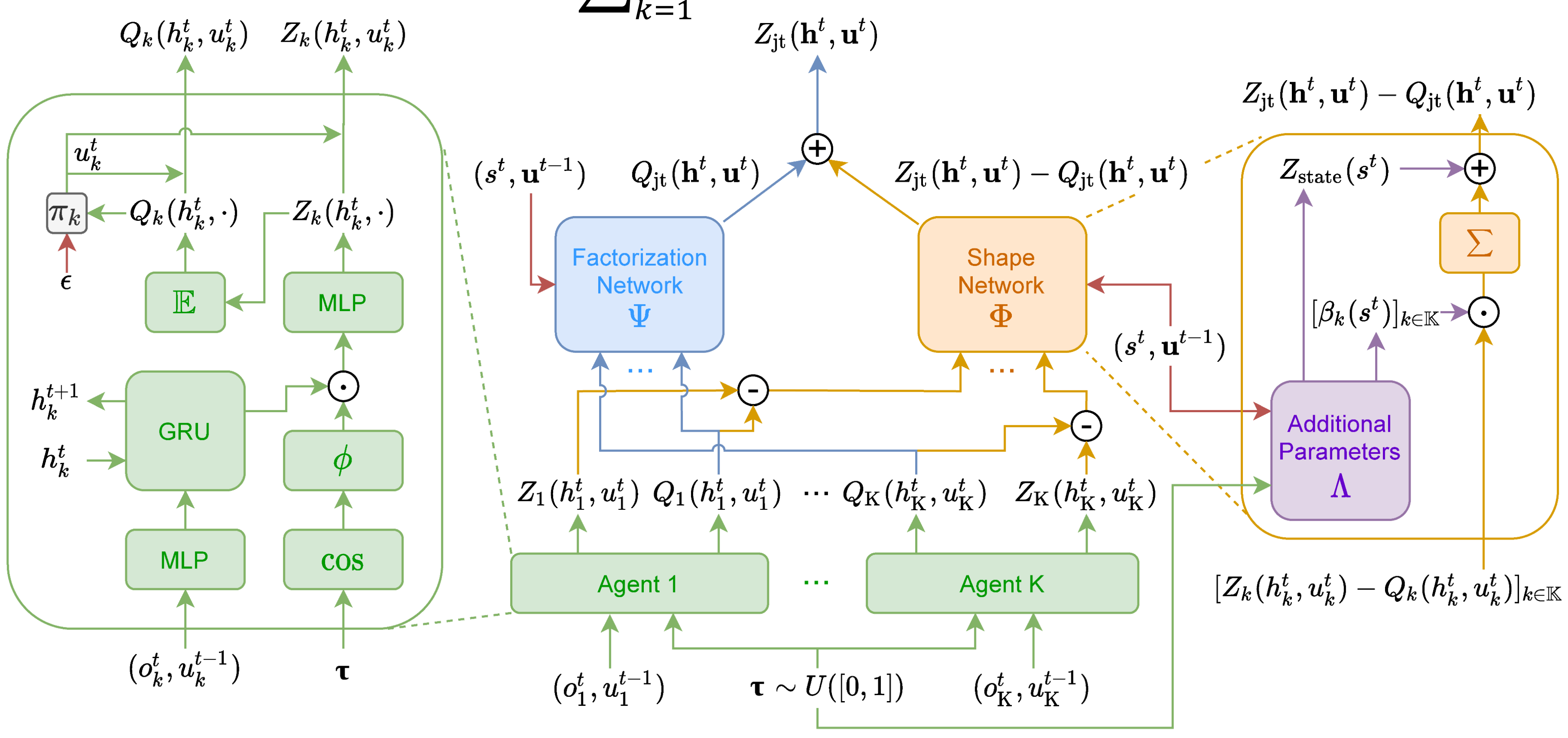
The DFAC Framework

- **DFAC Framework and Mean-Shape Decomposition**
  The naive generalization of the distributional form of IGM does not satisfy IGM in general. Thus, we introduced the mean-shape decomposition to separate the approximation of the mean and the shape of the return distribution:

$$\begin{aligned} Z_{\mathrm{jt}} &= \mathbb{E}[Z_{\mathrm{jt}}] + (Z_{\mathrm{jt}} - \mathbb{E}[Z_{\mathrm{jt}}]) = Z_{\mathrm{mean}} + Z_{\mathrm{shape}}, \text{where} \\ Z_{\mathrm{mean}} &= \Psi(s, Q_1(h_1,u_1), \ldots, Q_{\mathrm{K}}(h_{\mathrm{K}},u_{\mathrm{K}})) \\ Z_{\mathrm{shape}} &= \Phi(s, Z_1(h_1,u_1), \ldots, Z_{\mathrm{K}}(h_{\mathrm{K}},u_{\mathrm{K}})) \\ &= Z_{\mathrm{state}}(s) + \sum_{k=1}^{\mathrm{K}} \beta_k(s)(Z_k(h_k,u_k) - Q_k(h_k,u_k)) \qquad (1) \end{aligned}$$

- **A Practical Implementation with Quantile Mixture**
  The factorization network Ψ can be any expected value function factorization method, while the shape network Φ can be approximated by a quantile mixture:

$$\Phi(\tau) = F^{-1}_{\mathrm{state}}(s;\tau) + \sum_{k=1}^{\mathrm{K}} \beta_k(s)(F^{-1}_k(h_k,u_k;\tau) - Q_k(h_k,u_k)) \qquad (2)$$

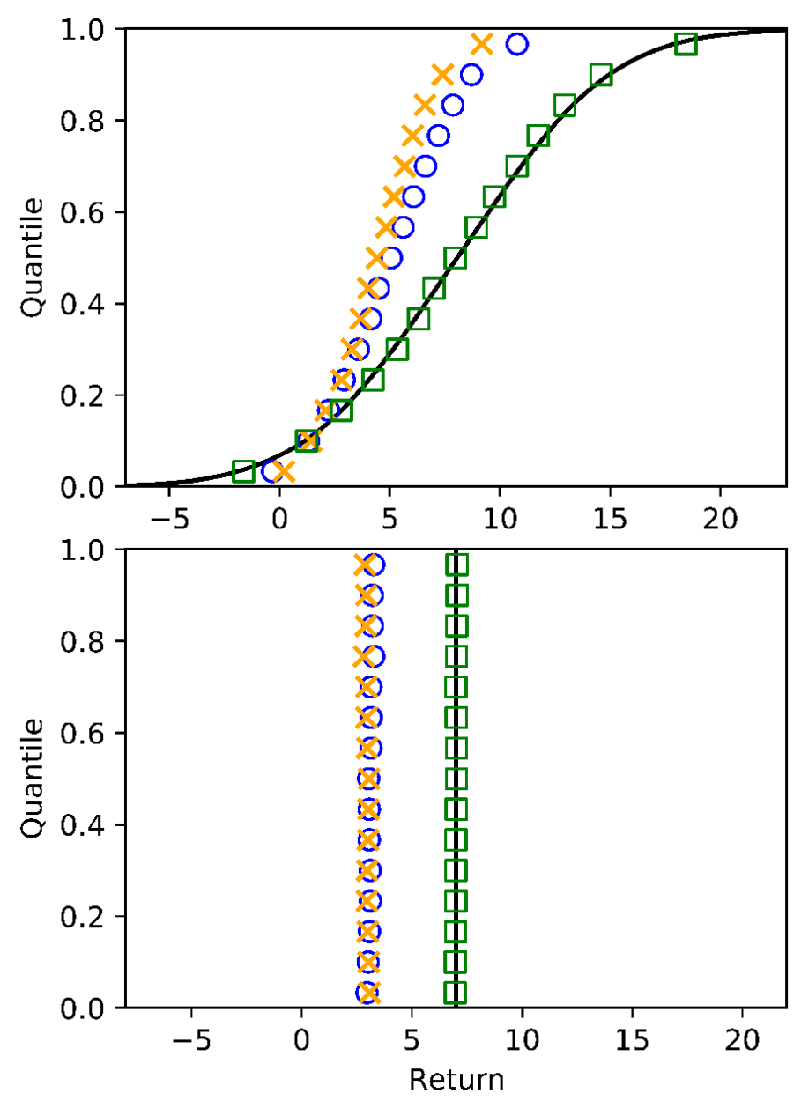


$Q_k(h_k^t,u_k^t)$
$Z_k(h_k^t,u_k^t)$
u_k^t
π_k
$Q_k(h_k^t,\cdot)$
$Z_k(h_k^t,\cdot)$
ϵ
$\mathbb{E}$
MLP
h_k^{t+1}
h_k^t
GRU
ϕ
MLP
cos
(o_k^t,u_k^{t-1})
τ
$Z_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t)$
$(s^t,\mathbf{u}^{t-1})$
$Q_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t)$
$Z_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t) - Q_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t)$
Factorization Network
Ψ
Shape Network
Φ
$(s^t,\mathbf{u}^{t-1})$
$Z_{\mathrm{state}}(s^t)$
Σ
$[\beta_k(s^t)]_{k\in\mathbb{K}}$
Additional Parameters
Λ
$Z_1(h_1^t,u_1^t)$ $Q_1(h_1^t,u_1^t)$ $\cdots$ $Q_{\mathrm{K}}(h_{\mathrm{K}}^t,u_{\mathrm{K}}^t)$ $Z_{\mathrm{K}}(h_{\mathrm{K}}^t,u_{\mathrm{K}}^t)$
Agent 1
Agent K
(o_1^t,u_1^{t-1})
$\tau\sim U([0,1])$
$(o_{\mathrm{K}}^t,u_{\mathrm{K}}^{t-1})$
$[Z_k(h_k^t,u_k^t) - Q_k(h_k^t,u_k^t)]_{k\in\mathbb{K}}$

(Fig.1) The architecture of the DFAC framework

Experiment Results

We generalize the two baselines: VDN and QMIX to their distributional variant: DDN and DMIX, respectively. The results showed that DDN and DMIX can **successfully factorize the joint return distribution** and **achieve outstanding performance** in SMAC:



(Fig.2) the Learned Factorization

Map	IQL	VDN	QMIX	DIQL	DDN	DMIX
6h_vs_8z	0.00%	0.00%	8.81%	0.00%	**83.52%**	68.75%
3s5z_vs_3s6z	7.67%	90.91%	65.06%	29.83%	**94.60%**	90.62%
MMM2	69.32%	87.78%	92.33%	83.52%	**97.44%**	95.17%
27m_vs_30m	1.70%	64.20%	86.08%	12.50%	**94.60%**	86.08%
corridor	83.10%	85.23%	4.26%	92.05%	**95.45%**	90.06%
6h_vs_8z	13.96	15.49	14.02	14.98	**19.32**	17.81
3s5z_vs_3s6z	15.48	19.77	20.06	17.42	20.68	**20.78**
MMM2	17.47	19.32	19.45	19.21	**21.06**	19.69
27m_vs_30m	13.95	18.49	19.46	15.16	**19.72**	19.40
corridor	19.30	19.38	13.44	19.57	**19.97**	19.61

(Table.1) Win rates and average scores in SMAC

Background

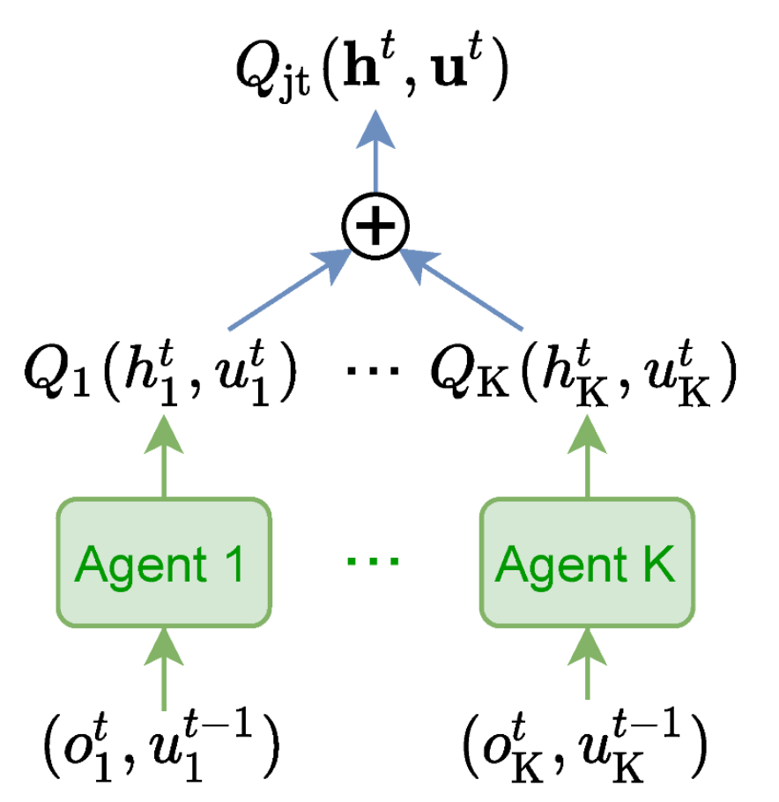
- **Value-based Learning Methods for Fully Cooperative MARL**
  Independent Q-Learning (IQL) is the simplest value-based learning method for MARL, where each agent attempts to maximize the total rewards separately. This causes unstationarity due to the changing policies of the other agents and may not converge. Thus, value function factorization methods are introduced to enable centralized training of factorizable tasks based on the **IGM (Individual-Global-Max) condition**, where optimal individual actions result in the optimal joint action of the group of agents:

$$\mathrm{argmax}_{\mathbf{u}} Q_{\mathrm{jt}}(\mathbf{h},\mathbf{u}) = \begin{pmatrix} \mathrm{argmax}_{u_1} Q_1(h_1,u_1) \\ \vdots \\ \mathrm{argmax}_{u_{\mathrm{K}}} Q_{\mathrm{K}}(h_{\mathrm{K}},u_{\mathrm{K}}) \end{pmatrix} \qquad (3)$$

The previous Value Decomposition Network (VDN) and QMIX methods assume additivity and monotonicity, respectively, to simplify the tasks:

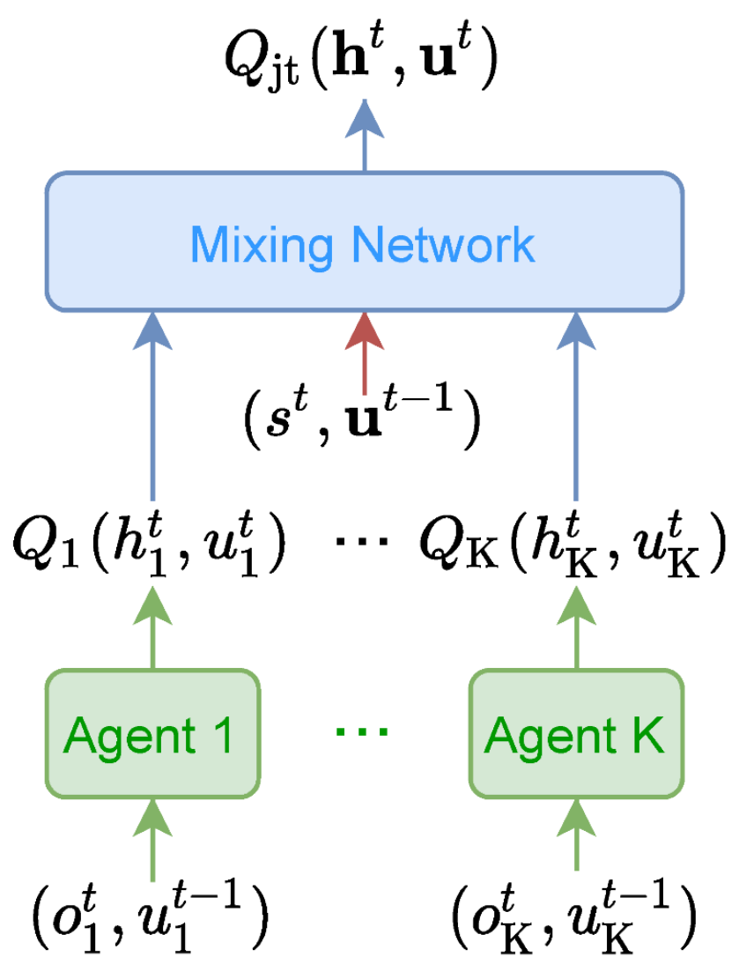
(Additivity)
$$Q_{\mathrm{jt}}(\mathbf{h},\mathbf{u}) = \textstyle\sum_{k=1}^{\mathrm{K}} Q_k(h_k,u_k) \qquad (4)$$

(Monotonicity)
$$Q_{\mathrm{jt}}(\mathbf{h},\mathbf{u}) = M(Q_1(h_1,u_1),\ldots,Q_{\mathrm{K}}(h_{\mathrm{K}},u_{\mathrm{K}}))$$
$$\text{where } \frac{\partial M}{\partial Q_k} \geq 0, \forall k \in \{1,\ldots,\mathrm{K}\} \qquad (5)$$



$Q_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t)$
$Q_1(h_1^t,u_1^t)$
$Q_{\mathrm{K}}(h_{\mathrm{K}}^t,u_{\mathrm{K}}^t)$
Agent 1
Agent K
(o_1^t,u_1^{t-1})
$(o_{\mathrm{K}}^t,u_{\mathrm{K}}^{t-1})$

(Fig.3) VDN



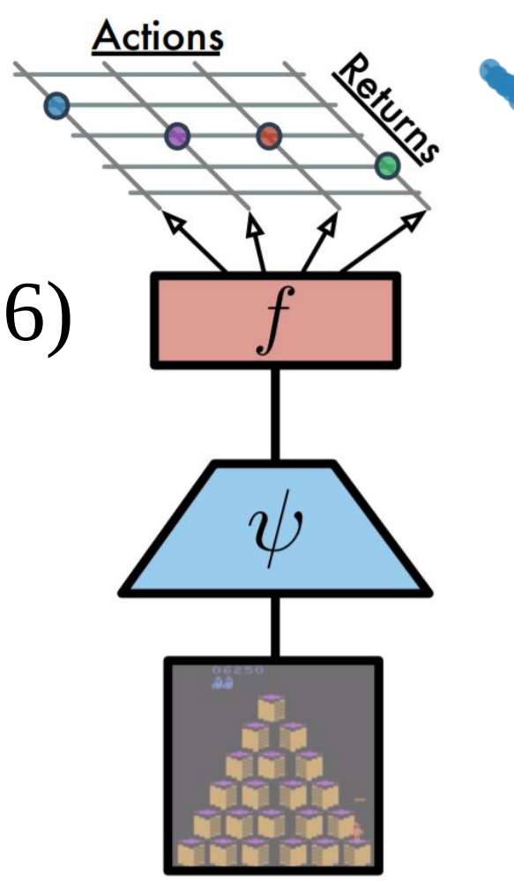
$Q_{\mathrm{jt}}(\mathbf{h}^t,\mathbf{u}^t)$
Mixing Network
$(s^t,\mathbf{u}^{t-1})$
$Q_1(h_1^t,u_1^t)$
$Q_{\mathrm{K}}(h_{\mathrm{K}}^t,u_{\mathrm{K}}^t)$
Agent 1
Agent K
(o_1^t,u_1^{t-1})
$(o_{\mathrm{K}}^t,u_{\mathrm{K}}^{t-1})$

(Fig.4) QMIX

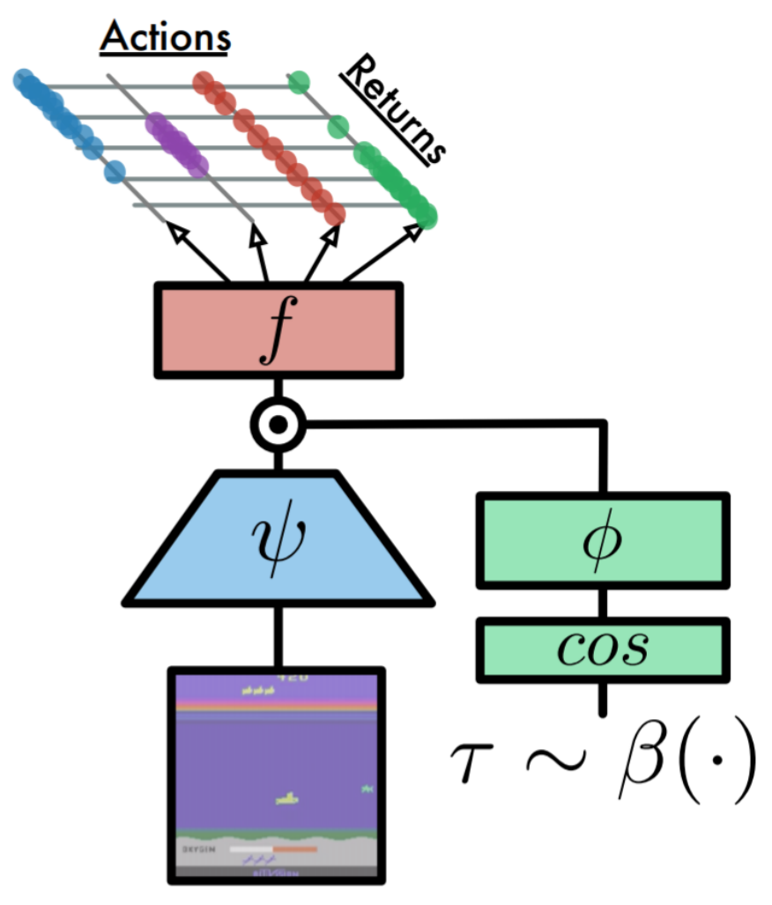
- **Distributional Reinforcement Learning**
  Distributional RL methods have been proved empirically to outperform expected RL methods in various single-agent RL (SARL) domain. The distributional Bellman operator T^π is proved to have a contraction in p-Wasserstein distance $W_p, \forall p \in [1,\infty)$:

$$T^\pi Z(s,u) \overset{D}{=} R(s,u) + \gamma Z(s',u')$$
$$W_p(X,Y) = \left(\int_0^1 |F_X^{-1}(\tau) - F_Y^{-1}(\tau)| d\tau\right)^{1/p} \qquad (6)$$

Implicit Quantile Network (IQN) is by far the most light-weight distributional RL algorithm. It models the quantile function of the return distribution, and can efficiently approximate the expectation by inverse distribution sampling$[\tau_i \sim U([0,1])]_{i=1}^{N}$:



(Fig.5) DQN



(Fig.6) IQN

$$Q(s,u) = \mathbb{E}[Z(s,u)] = \int_0^1 F^{-1}(s,u;\tau)d\tau \approx \frac{1}{N}\sum_{i=1}^{N} F^{-1}(s,u;\tau_i) \qquad (7)$$

Contributions

We introduced DFAC for integrating distributional RL with MARL domain. We first proposed **mean-shape decomposition** procedure to ensure the IGM condition holds for all factorizable tasks. Then, we proposed the use of **quantile mixture** to implement the mean-shape decomposition in a computationally friendly manner. In order to validate the effectiveness of DFAC, we presented experimental results performed on all Super Hard scenarios in SMAC for a number of MARL baseline methods as well as their DFAC variants. The results show that DDN and DMIX outperform VDN and QMIX. DFAC can be extended to more value function factorization methods and offers an interesting research direction for future endeavors. More experiment results are presented in the full paper.

More Information

For the full paper, references, and gameplay video, please scan the QR Code below:

Questions?

j3soon@gapp.nthu.edu.tw

Conferences

AAMAS2021 LONDON.UK GTC21

Acknowledgement

NVIDIA

(Fig.7) One of the maps in SMAC