ACM Symposium on Cloud Computing

# Proactive Look-Ahead Control of Transaction Flows for High-Throughput Payment Channel Network

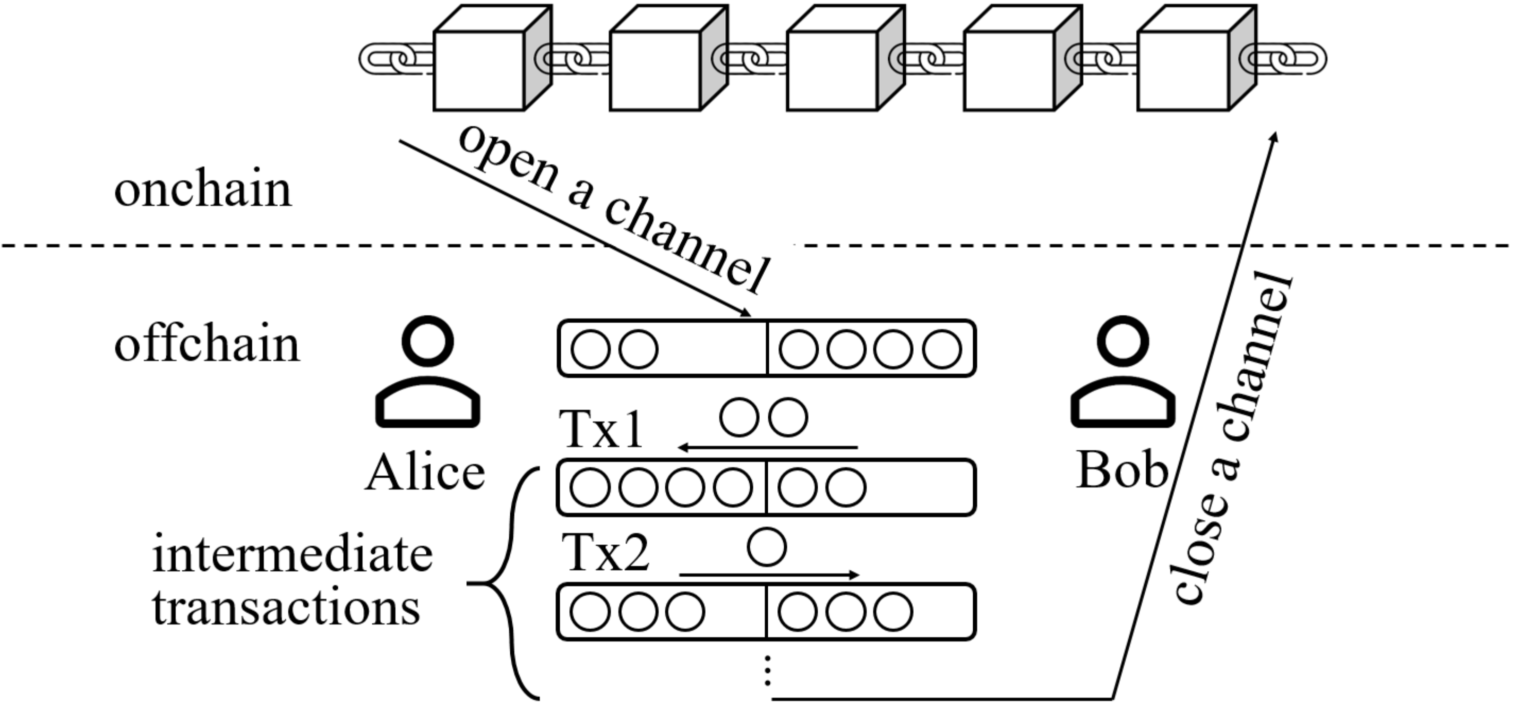
**Wuhui Chen**, Xiaoyu Qiu, Zicong Hong, Zibin Zheng,
Hong-Ning Dai, Jianting Zhang

中山大學
SUN YAT-SEN UNIVERSITY

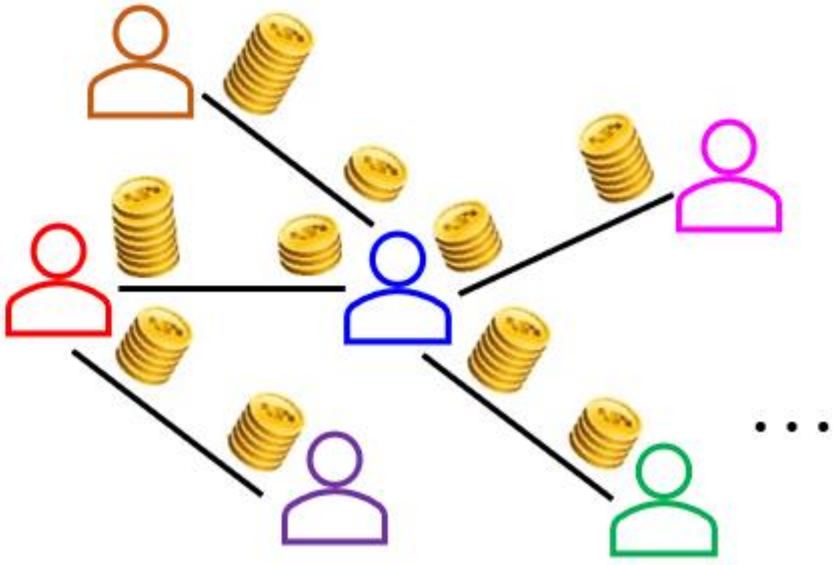
# ❑ Outlines

1. Background
2. Measurement
3. Algorithm Design
4. Experiments
5. Conclusion

## ❑ Payment Channel Network (PCN): a leading solution to scale blockchain

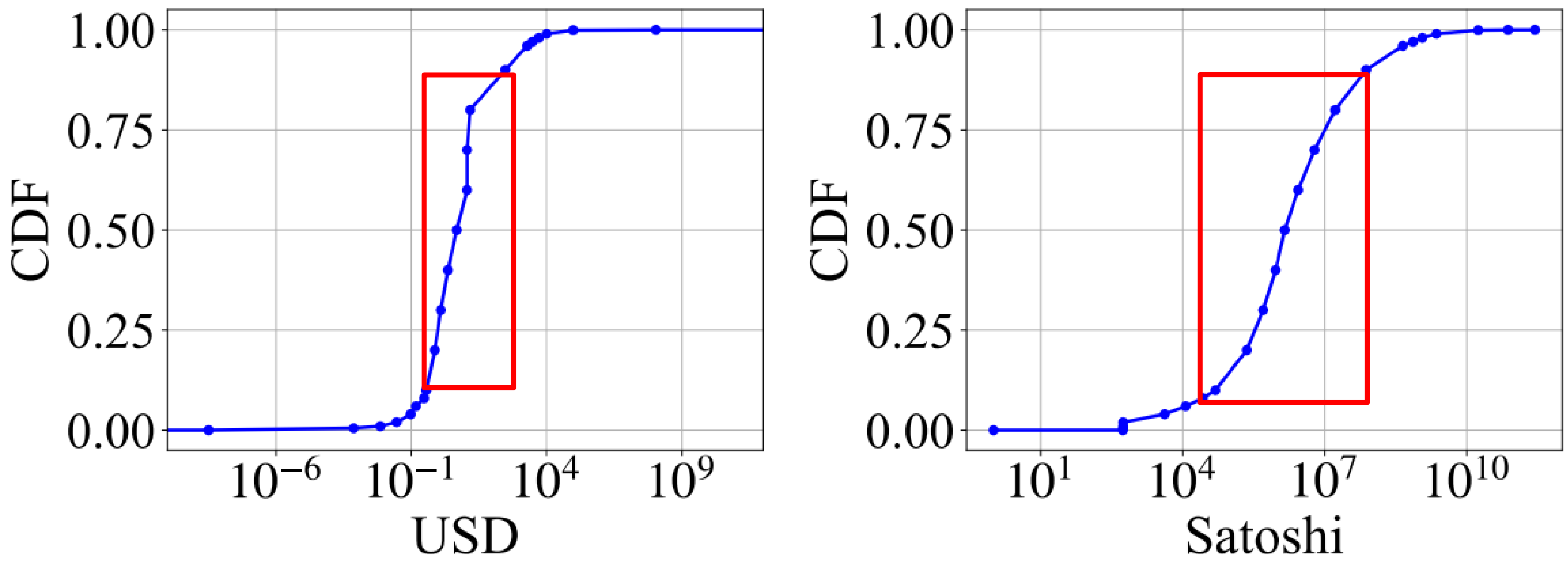



(a) How the payment channel works.



(b) A top example of PCN.

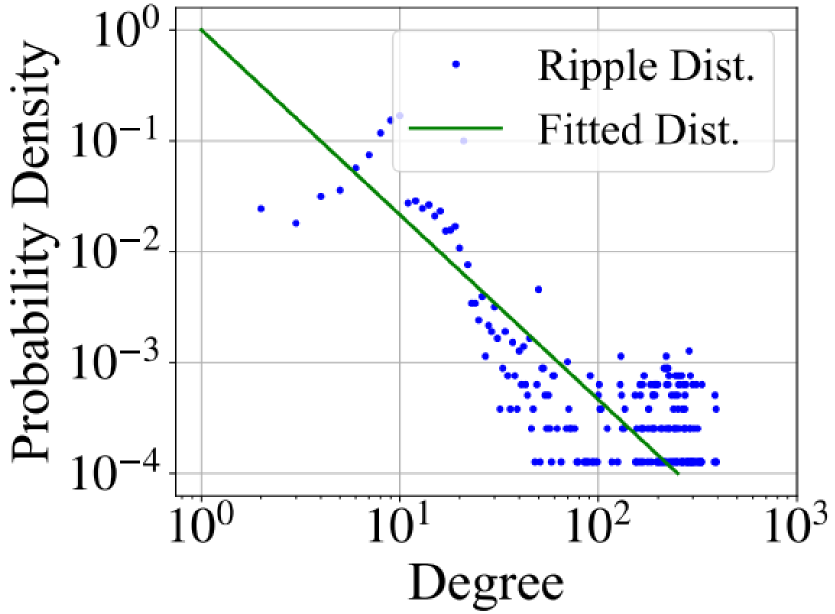
## ❑ Measurement #1: Distribution Analysis of PCN Transactions



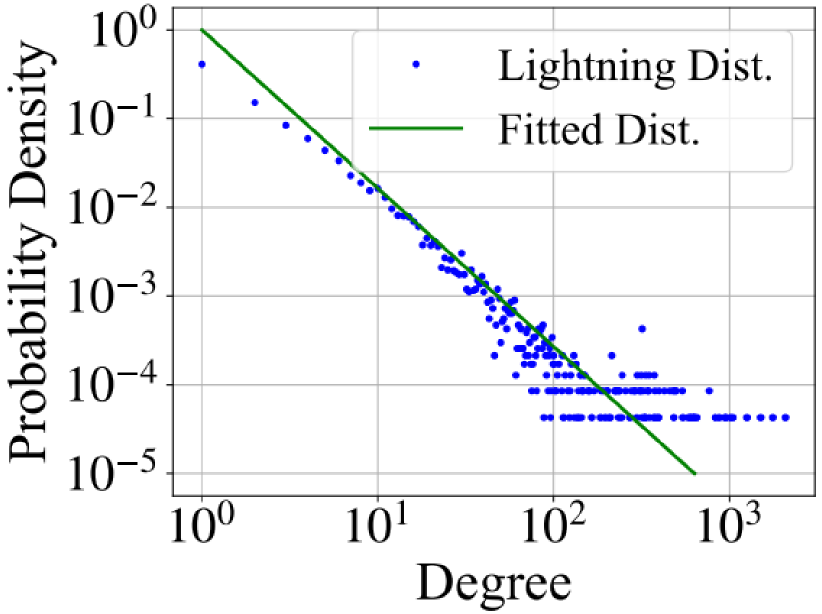

(a) Ripple (b) Bitcoin

**Insight #1**. The distribution of transaction amounts is highly **concentrated** and most values are distributed within a small range. This leads to a more traceable long-term throughput.

## ❑ Measurement #2: Analysis of Network Topology



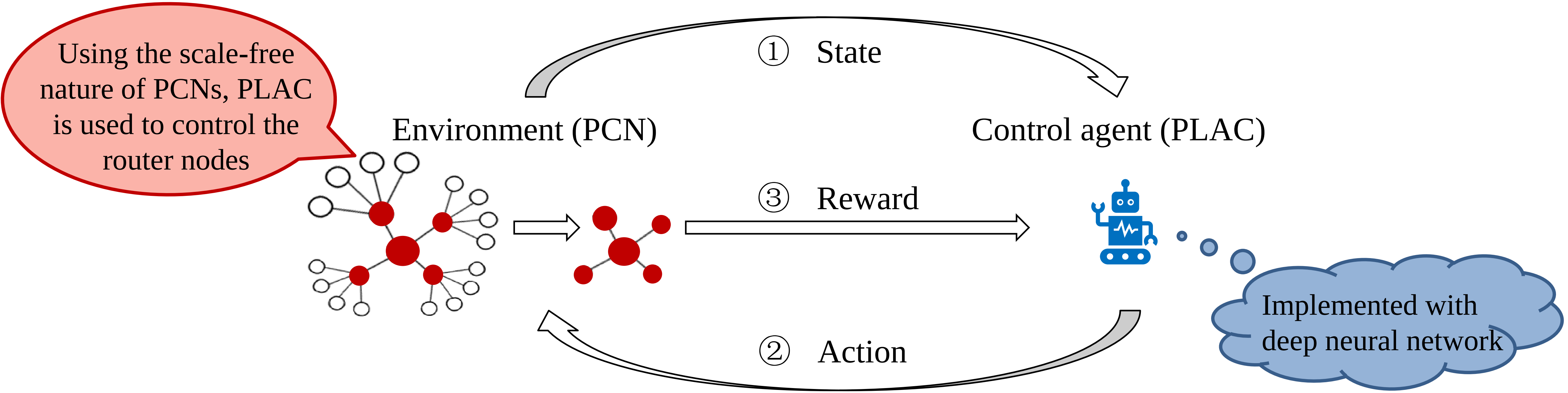

(a) Ripple




(b) Lightning

**Insight #2**. The degree distribution of PCN is similar to **scale-free networks**, suggesting that the key to scheduling transactions is the control of nodes with high degrees.

❑ **Proposed Solution: PLAC,** a deep reinforcement learning-based (DRL) transaction scheduling algorithm

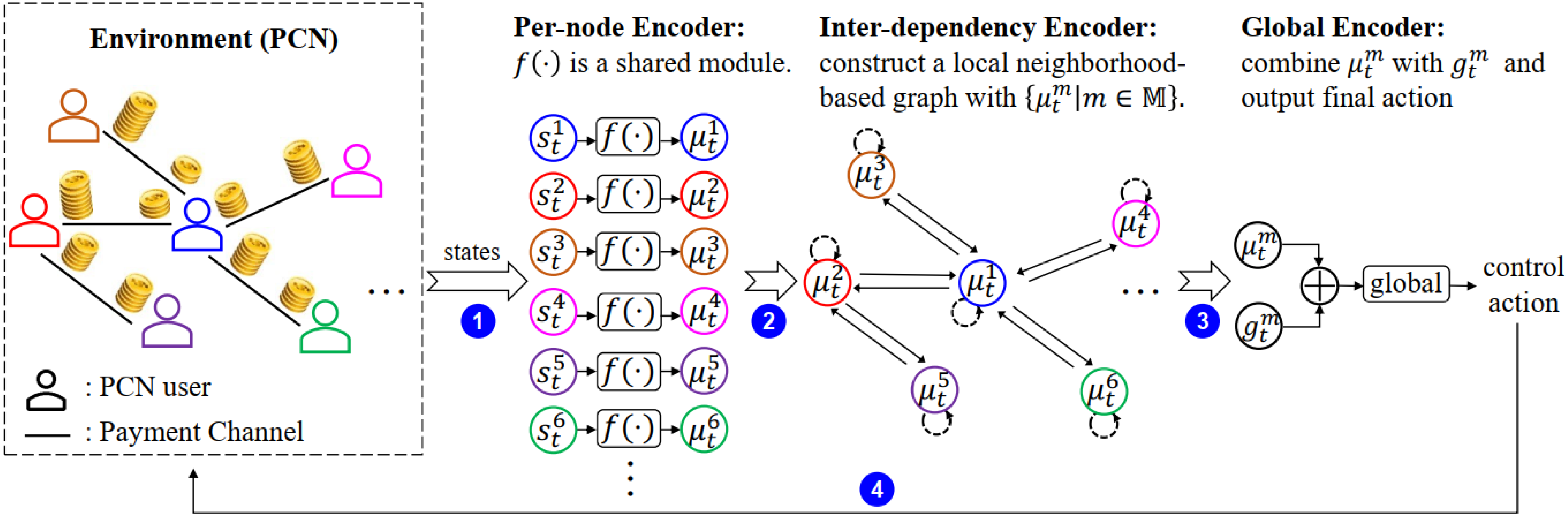



① **State**:
- topological structure of router nodes
- incoming transactions queued in router nodes
- past channel balances of a fixed window

② **Action**: the maximum transaction amount allowed through each channel

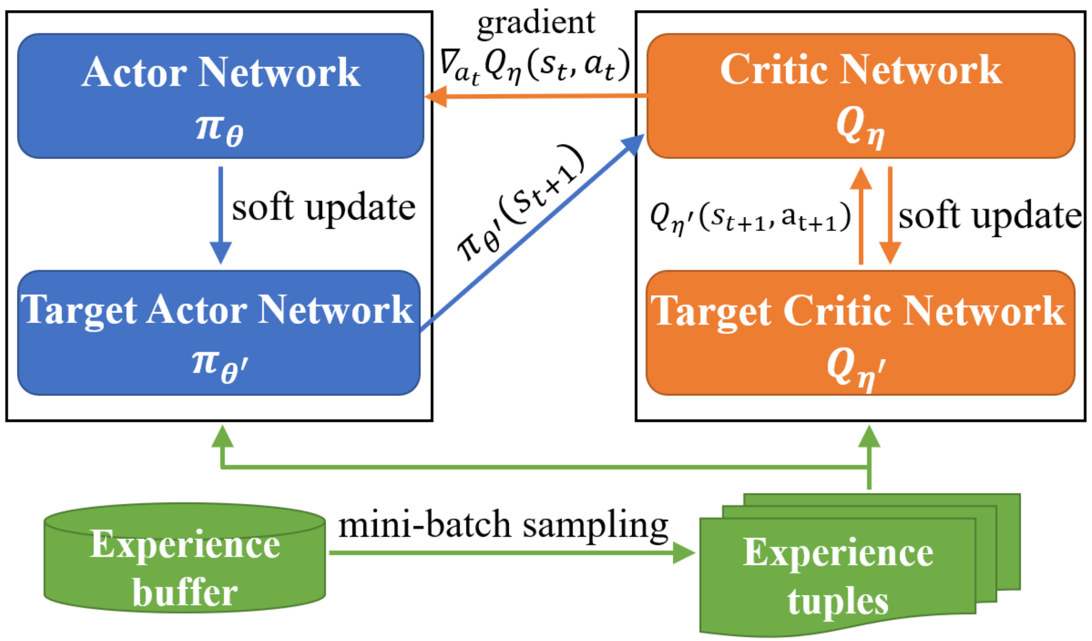
③ **Reward**: the number of successful transactions at current epoch

- **Network Design of PLAC:** a GNN-based model for graph-structured data



## ❑ Training methodology of PLAC

- We build PLAC with the famous actor-critic architecture.
- The **actor network** takes the state as input and outputs the action.
- The **critic network** is used to approximate the long-term reward function and provide guidance for actor-network's update;




① **Critic network update**:

$$\mathcal{L}(\eta) = \mathbb{E}\left[\left(Q_\eta(s_t, a_t) - \mathcal{T}Q_\eta(s_t, a_t)\right)^2\right],$$

where

$$\mathcal{T}Q_\eta(s_t, a_t) = r_t + \gamma \cdot \mathbb{E}\left[Q_\eta\left(s_{t+1}, \pi_\theta(s_{t+1})\right)\right].$$

② **Actor network update**:

$$\nabla_\theta J(\theta) = \mathbb{E}\left[\nabla_\theta Q_\eta(s, a)\Big|_{s=s_t, a=\pi_\theta(s_t)}\right].$$

## ❑ Training methodology of PLAC

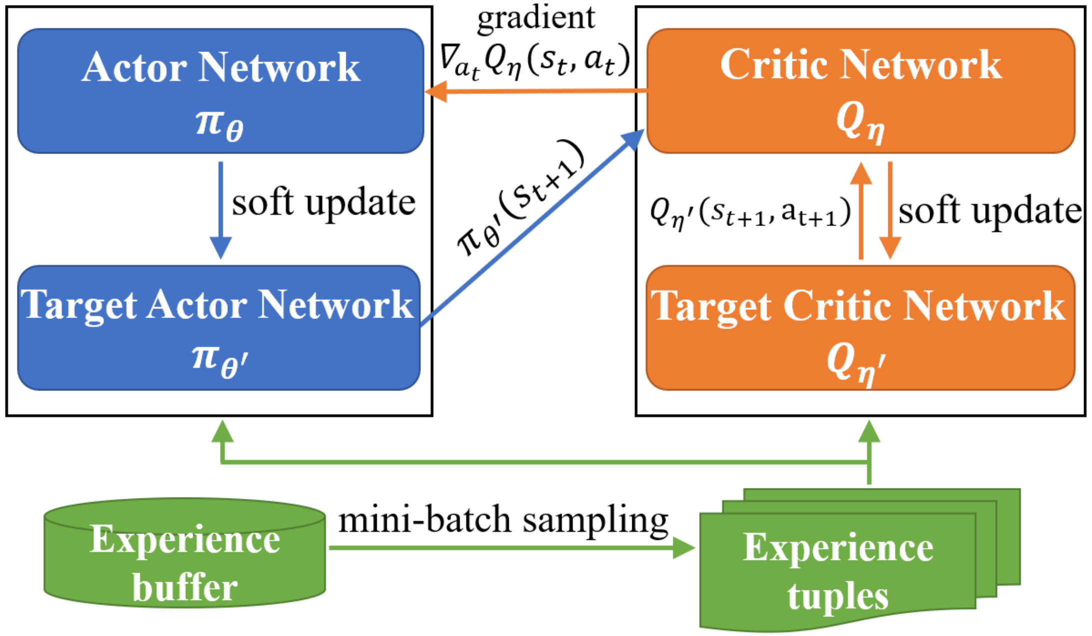
- To stabilize training, PLAC implements additional models for both actor and critic networks.
- The target value $\mathcal{T}Q_\eta(s_t, a_t)$ for critic network's update is rewritten as:

$$r_t + \gamma \cdot \mathbb{E}\left[Q_{\eta'}\left(s_{t+1}, \pi_{\theta'}(s_{t+1})\right)\right]$$

- The target network is updated using soft update:

$$\theta' = \tau\theta + (1-\tau)\theta',$$
$$\eta' = \tau\eta + (1-\tau)\eta',$$

where $\tau \ll 1$.

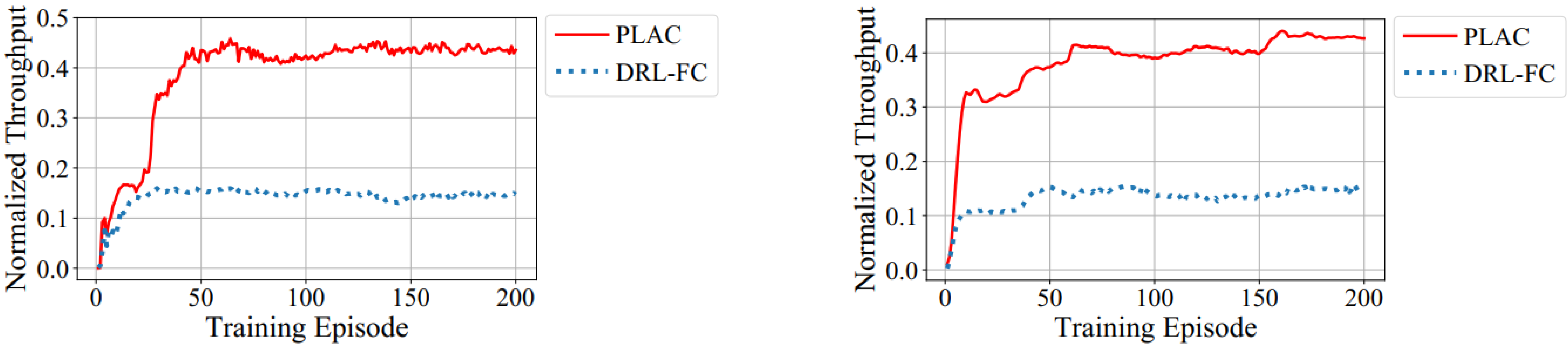


## ❑ Evaluation Setup

- Dataset:
  - Topology: Ripple on July 4, 2021; Lightning on December 30, 2020
  - Transaction
- Router node selection: top 40 nodes with the largest degrees
- Transaction generation: Sample from historical transactions
- Baselines:
  - Waterfilling
  - Flash
  - Shortest Path First (SPF)
  - DRL-FC
- Evaluation metric: the percentage of successful payments over all generated payment demands within a given time

# ❑ Evaluation Results

- Convergence Behavior

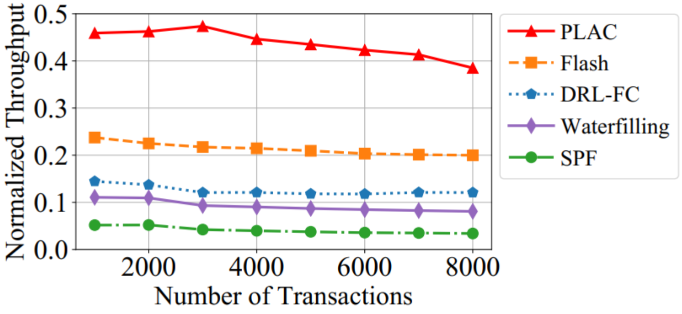



(a) Ripple (b) Lightning

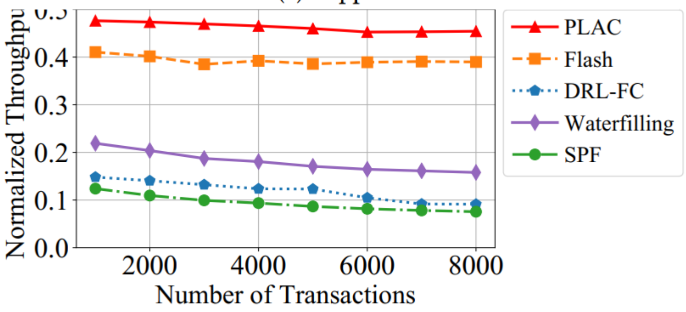
For Ripple and Lightning, PLAC achieves the normalized throughputs of about 45.9% and 47.6%, respectively. By contrast, DRL-FC only achieves the normalized throughputs of about 14.4% and 14.8% on Ripple and Lightning, respectively.

# ❑ Evaluation Results

- Performance with Different Evaluation Settings: PLAC improves the throughput by 6.6%–34.9% compared with the baselines.

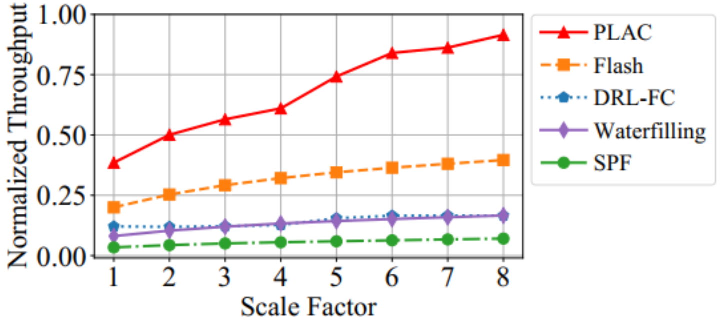



(a) Ripple

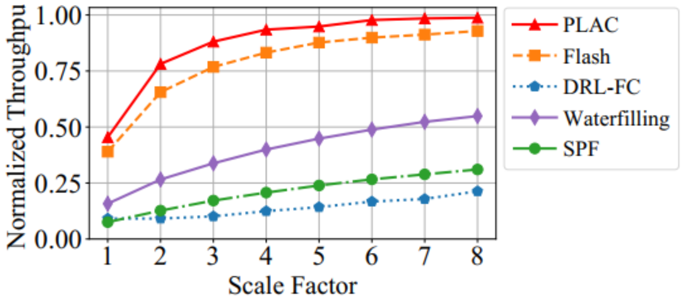



(b) Lightning

① Performance with different transaction load.

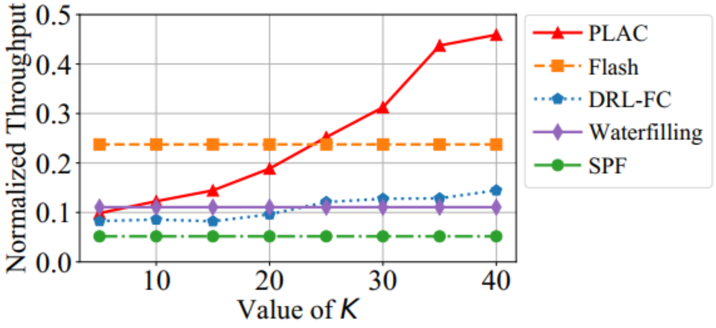



(a) Ripple

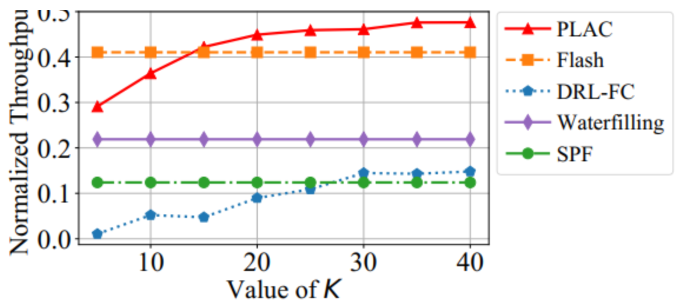



(b) Lightning

② Performance with different channel capacities.




(a) Ripple




(b) Lightning

③ Performance with different number of router nodes.

- **Conclusions:**
  - It is important to consider the long-term effect of transactions on the channel balances when scheduling PCN transactions.
  - The scale-free nature of PCN allows us to focus on nodes with high degrees when scheduling transactions.
  - By leveraging DRL and GNN, PLAC can learn a scheduling policy that leads to higher long-term throughput than baselines.

ACM Symposium on Cloud Computing

# Thank you!

SUN YAT-SEN UNIVERSITY