Xueying HAN, Mingxi XIE, Ke YU, Xiaohong HUANG, Zongpeng DU, Huijuan YAO, 2024. Combining graph neural network with deep reinforcement learning for resource allocation in computing force networks. *Frontiers of Information Technology & Electronic Engineering*, 25(5):701-712.
https://doi.org/10.1631/FITEE.2300009

Combining graph neural network with deep reinforcement learning for resource allocation in computing force networks

Key words: Computing force network; Routing optimization; Deep learning; Graph neural network; Resource allocation

Corresponding author: Ke YU
E-mail: yuke@bupt.edu.cn
ORCID: https://orcid.org/0000-0002-1158-1483

Motivation

1. In the future digital society, a significant number of computing devices will be dispersed at different sites close to users, and provide them with various personalized services through global networks. Computing force network (CFN) is proposed as a promising paradigm that leverages ubiquitous computing resources distributed in the network. However, accommodating network and computing resources jointly and efficiently is still a challenge.

2. In designing a resource allocation approach for CFN, it is of paramount importance to consider multi-dimensional resources, obtain optimization results in time, and solve the generalization problem of existing algorithms based on deep reinforcement learning.

Main idea

1. To optimize network resources and computing resources jointly in CFN, we propose a graph neural network (GNN) based deep reinforcement learning (DRL) architecture, which can obtain optimization decisions in a timely manner.

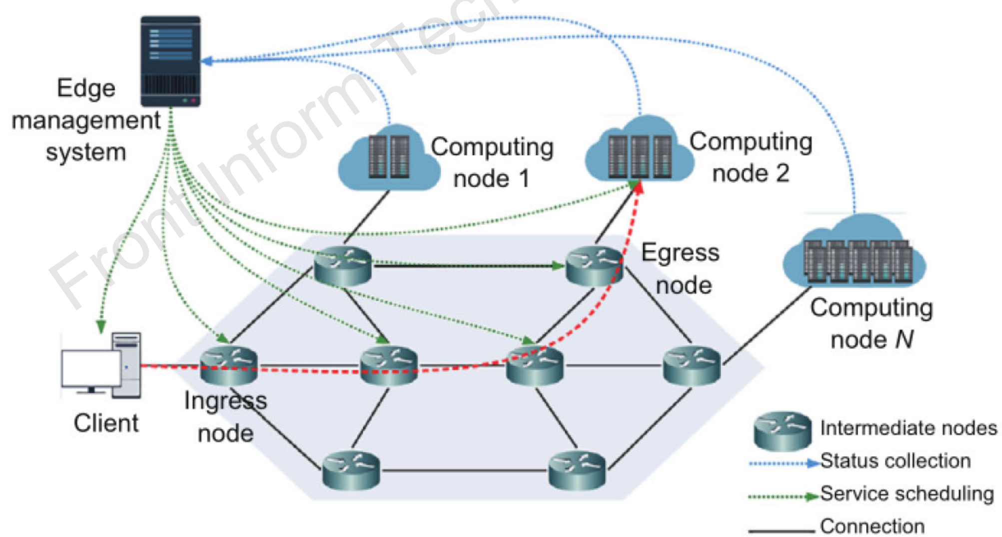
2. For the DRL agent which learns to maximize the objective function, we devise a multi-dimensional reward function that accommodates network and computing resources.

3. By taking advantage of GNN, we integrate the message passing neural network (MPNN) in the internal DRL neural network. The proposed method can operate and generalize over diverse network topologies, even if the structure changes.

Scenario

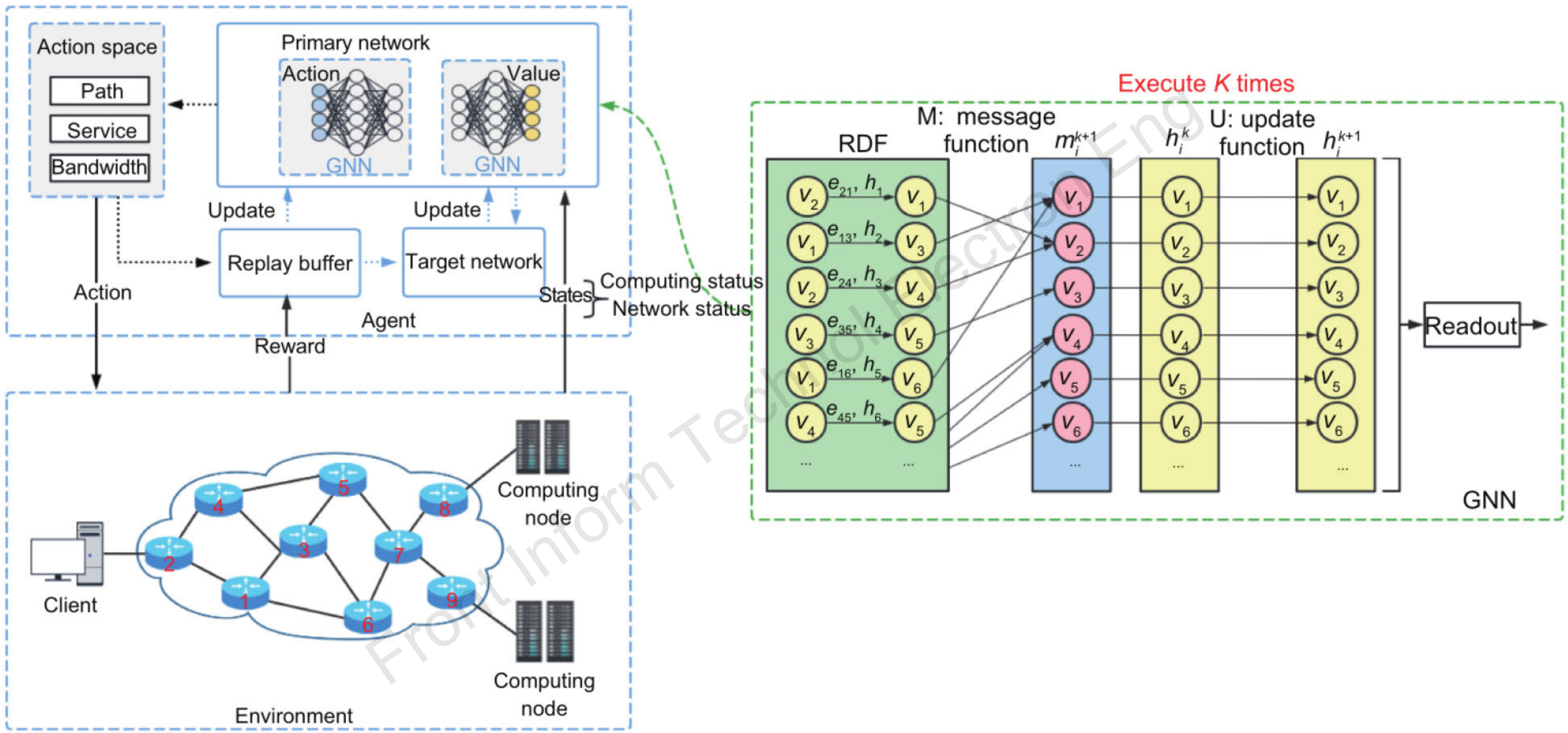
CFN aims to connect the ubiquitous resources distributed in the terminal, edge, and cloud to provide ultra-low-latency and real-time computing services for diverse applications across the network.

CFN consists of five entities: clients, computing nodes, ingress nodes, egress nodes, and intermediate nodes.



Architecture of the computing force network (CFN)

Framework



The graph neural network (GNN) based deep reinforcement learning (DRL) framework in the computing force network (CFN)

Method

1. To improve the expressiveness and generalization of the resource allocation approach, we propose to combine a deep Q-learning network (DQN) with GNN, in which hidden states can be introduced to learn edge features and can be adopted flexibly to reason different relations in links and nodes.

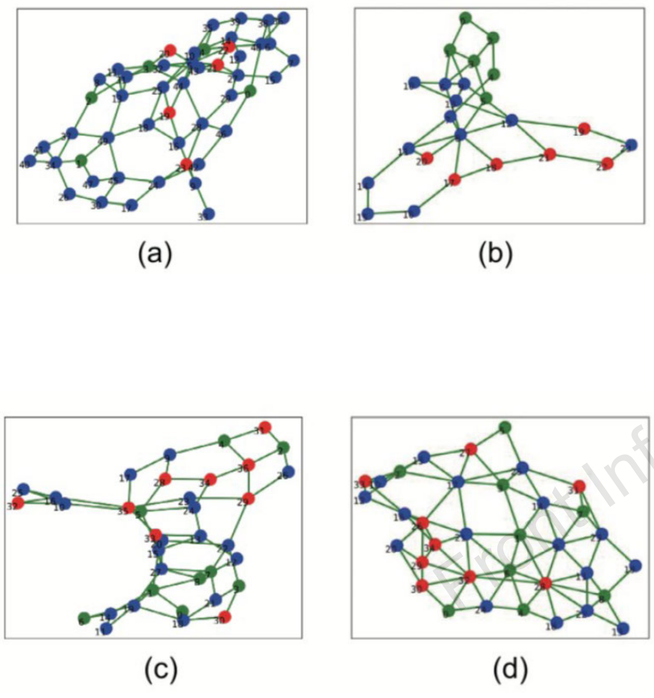
2. MPNN, as a general framework of GNN, can be adopted as the internal neural network in the DQN agent. We encode hidden states in MPNN, denoted as $h_i=(x_1, x_2, ..., x_N)$. The values x_1 to x_3 represent network states and the action of bandwidth allocation. The computing resources are stored in x_4 to x_6, including the computing type, load, and node assigned for the service request.

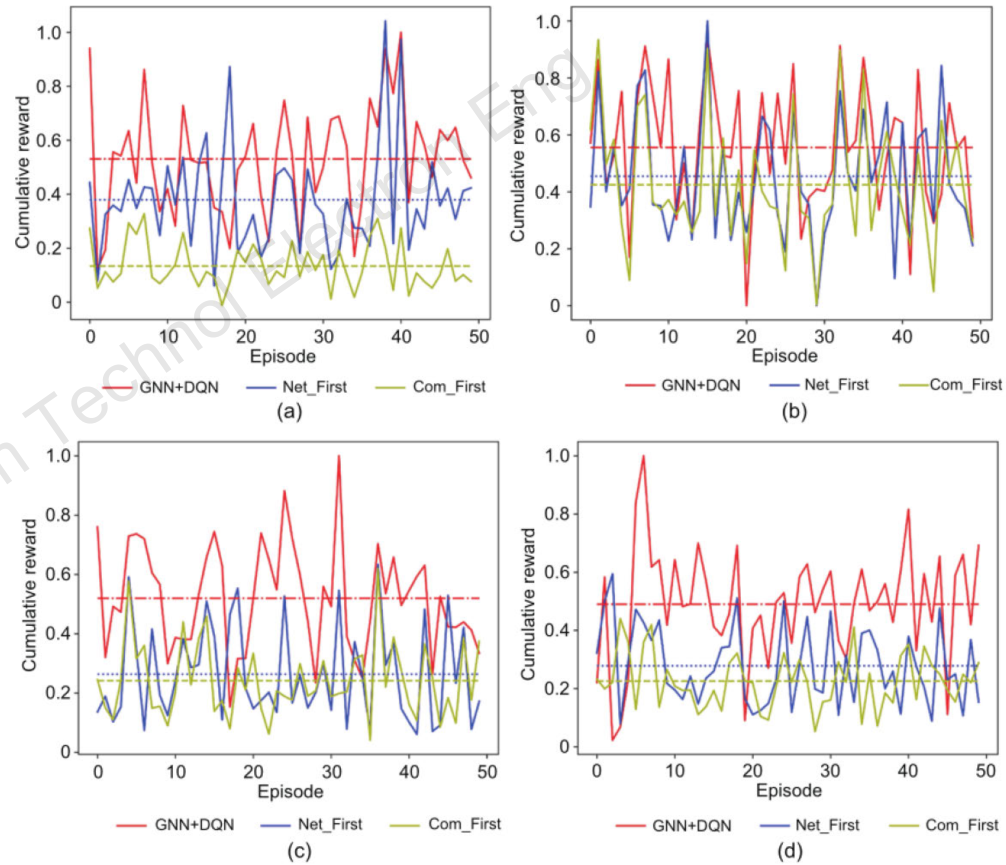
Method

3. The design details of the proposed DRL agent are as follows:

- Environment: The environment refers to the data plane in CFN.
- Agent: The agent serves as the decision-maker, interacts with the environment, and observes the consequences by trial and error. In a centralized CFN, the control plane can work as the agent to collect states, make decisions, and deploy an action to the data plane.
- State: The state refers to the state of links and nodes.
- Action: The action space in the CFN scenario involves path selection, service scheduling (choosing eligible computing nodes), and allocation of bandwidth resources for the traffic.
- Reward: The reward function is devised by considering optimizing both network resources (bandwidth) and computing resources (computing power).

Major results





Performance comparison of the cumulative reward with different methods:
(a) Germany50; (b) Geant2; (c) Cost266; (d) India35

Conclusions

1. Faced with the challenge of resource allocation in a computing force network (CFN), this paper proposes a graph neural network (GNN) based deep reinforcement learning (DRL) architecture to accommodate network resources and computing resources jointly and efficiently.

2. In the proposed architecture, we leverage the model-free DRL framework to devise a multi-dimensional reward function that integrates network and computing resources. We combine GNN with the DRL agent to improve the method’s generalization. Finally, a trained agent can obtain optimization decisions in real time and can adapt well to the updated topology.