Asynchronous Deep Reinforcement Learning for Data-Driven Task Offloading in MEC-Empowered Vehicular Networks

Penglin Dai*, Kaiwen Hu*, Xiao Wu*, Huanlai Xing*, and Zhaofei Yu†
Email:{penglindai@swjtu.edu.cn, kaiwenhu swjtu@163.com, wuxiaohk@gmail.com, hxx@home.swjtu.edu.cn, yzf714@126.com}
* School of Information Science and Technology, Southwest Jiaotong University, Chengdu, 611756, China
† School of Electronics Engineering and Computer Science, Peking University, Beijing 100871, China

CONTENT

1 Introduction

2 System Model

3 Problem Formulation

4 Algorithm

5 Experiment

The characteristics of data-driven task

- Traffic data sensed by vehicle
- Computation-intensive
- Delay-sensitive

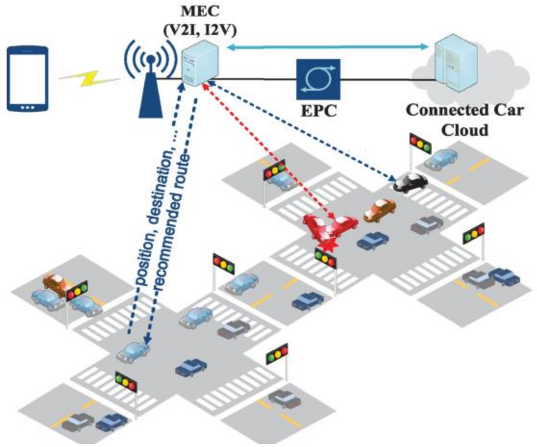


Fig. 1 Data-driven task in vehicular networks

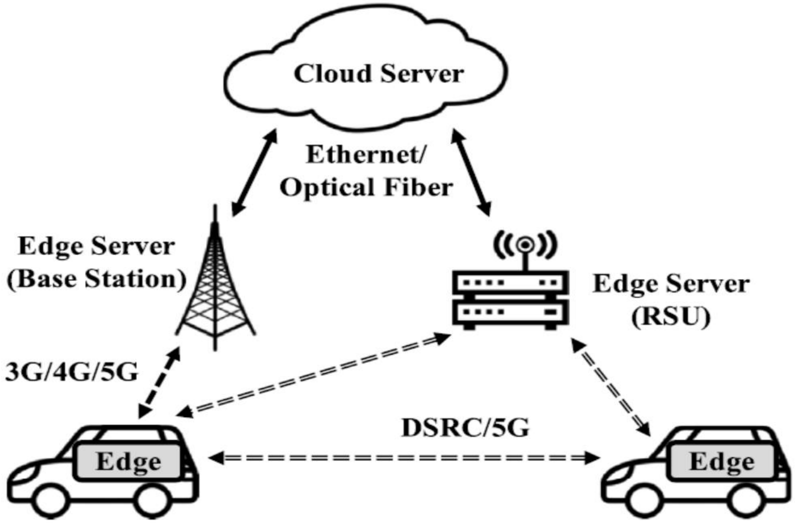


Fig. 2 Autonomous driving edge computing architecture[1]

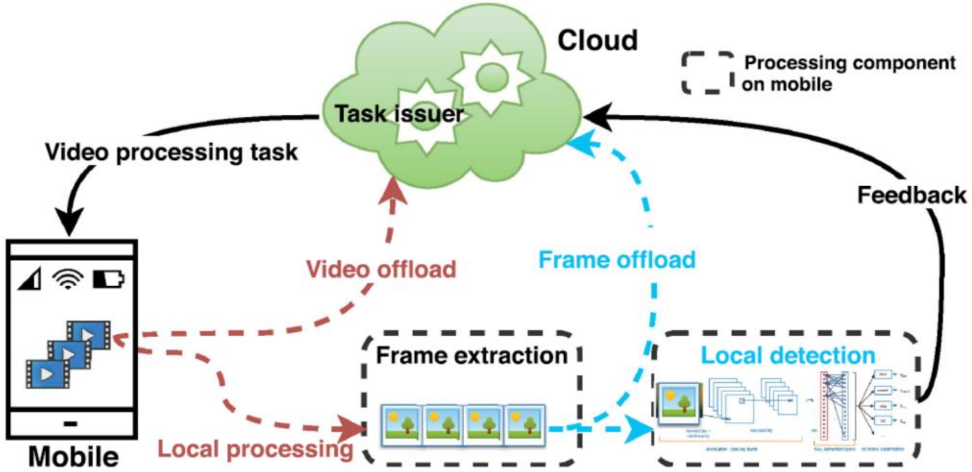


Fig. 3 An Edge Computing Platform for VideoCrowd processing[2]

[1] S. Liu, L. Liu, J. Tang, B. Yu, Y. Wang and W. Shi, "Edge Computing for Autonomous Driving: Opportunities and Challenges," in *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1697-1716, Aug. 2019, doi: 10.1109/JPROC.2019.2915983.

[2] Z. Lu, et.al, "CrowdVision: A Computing Platform for Video Crowdprocessing Using Deep Learning," in *IEEE Transactions on Mobile Computing*, vol. 18, no. 7, pp. 1513-1526, 1 July 2019.

1.1 MEC-based Architecture

Cloud Computing

1. Support wide range of services
2. Powerful resources
3. Cannot support real-time ITS applications

Mobile edge computing

1. Deploy at network edge and handle local task process
2. Reduces wired bandwidth
3. Support location-aware, low-latency, mobility-aware services

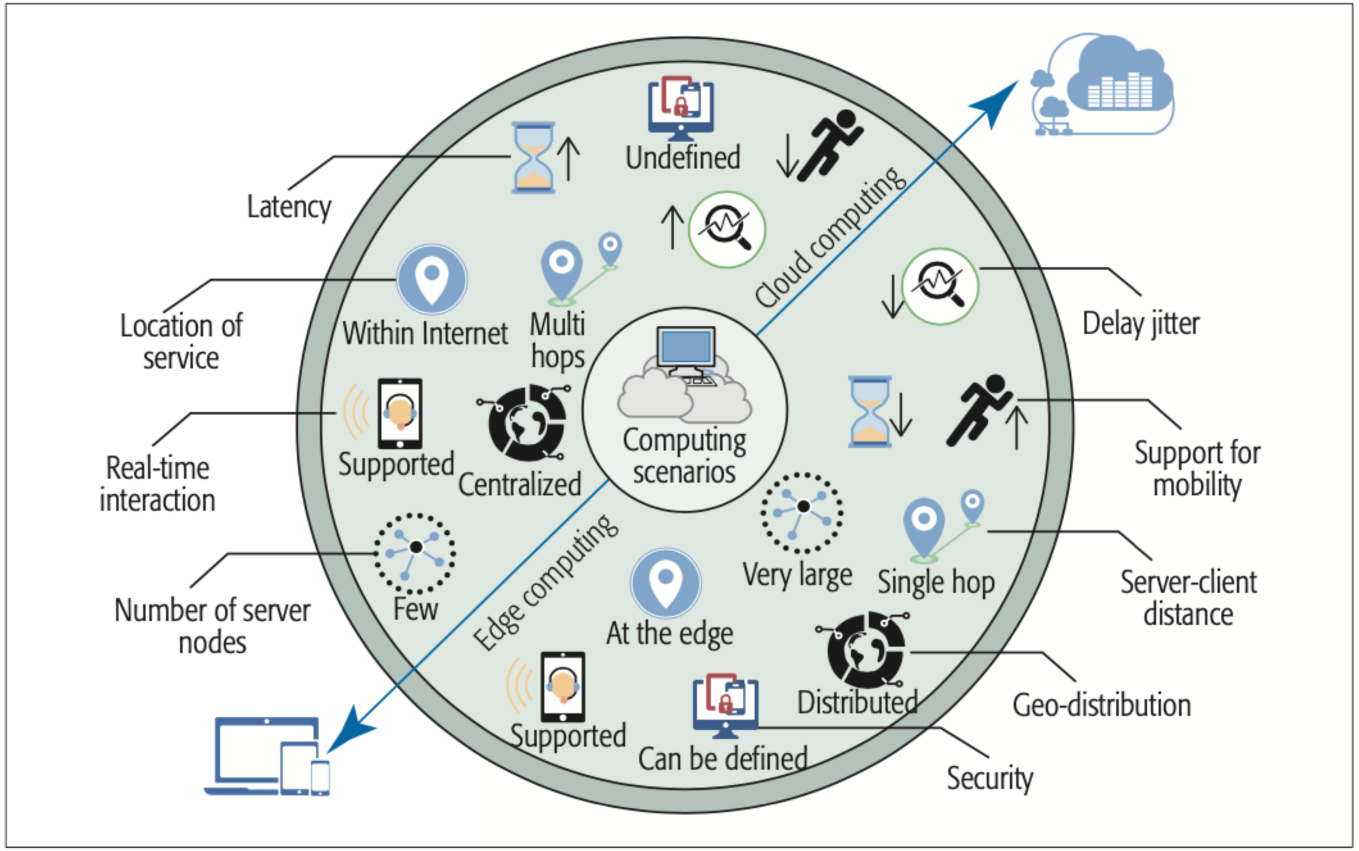


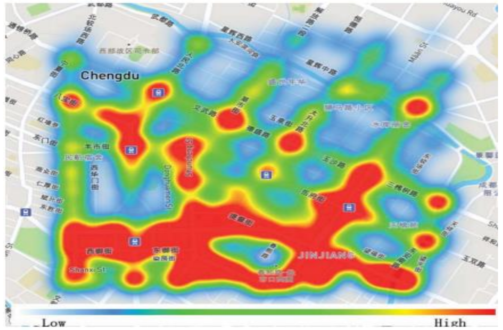
Fig.4 Edge computing vs. Cloud Computing [3]

[3] S. Garg, A. Singh, S. Batra, N. Kumar and L. T. Yang, "UAV-Empowered Edge Computing Environment for Cyber-Threat Detection in Smart Vehicles," in *IEEE Network*, vol. 32, no. 3, pp. 42-51, May/June 2018, doi: 10.1109/MNET.2018.1700286.

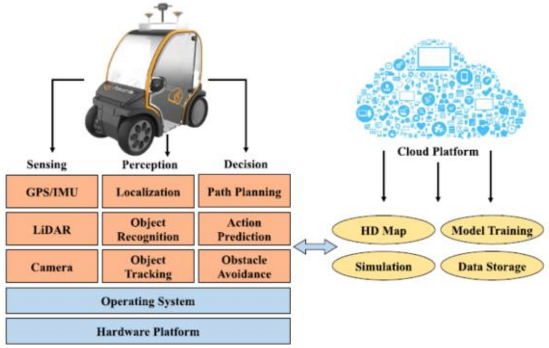
Challenge of computation offloading in vehicular networks

- **Service Characteristics**

High mobility of vehicles



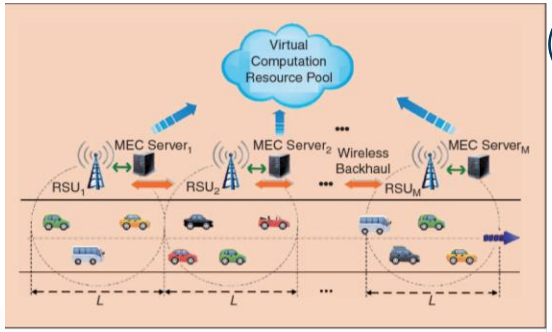
Dynamic distribution of vehicle density



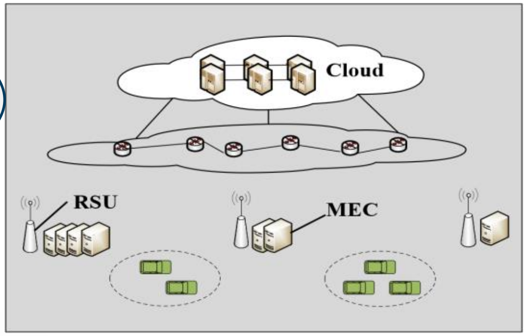
Computation-intensive and delay-sensitive

- **MEC**

Unbalanced workload and serious service delay



Limited service range



Heterogeneous caching/communication/computation resources

The contributions of this paper

1. Investigate a service scenario of MEC-based vehicular crowdsourcing, which exploits the heterogeneous resources of mobile vehicles, MEC servers and cloud.

2. Formulate data-driven task offloading (DTO) problem as a mixed-integer programming model by jointly considering offloading decision and resource allocation.

3. First design offloading algorithm based on asynchronous deep reinforcement learning and derive the optimal resource allocation base on convex theory.

4. Build a comprehensive simulation model and implement the proposed algorithm.

- Application layer
  - ✓ Traffic data collected by crowdsourcing vehicles
  - ✓ Divided into multiple subtasks
- Vehicular Layer
  - ✓ Crowdsourcing vehicles
  - ✓ Computing vehicles
- MEC Layer
  - ✓ Local scheduler
  - ✓ Computation server
- Cloud Layer
  - ✓ Communication via BS
  - ✓ Unlimited resource

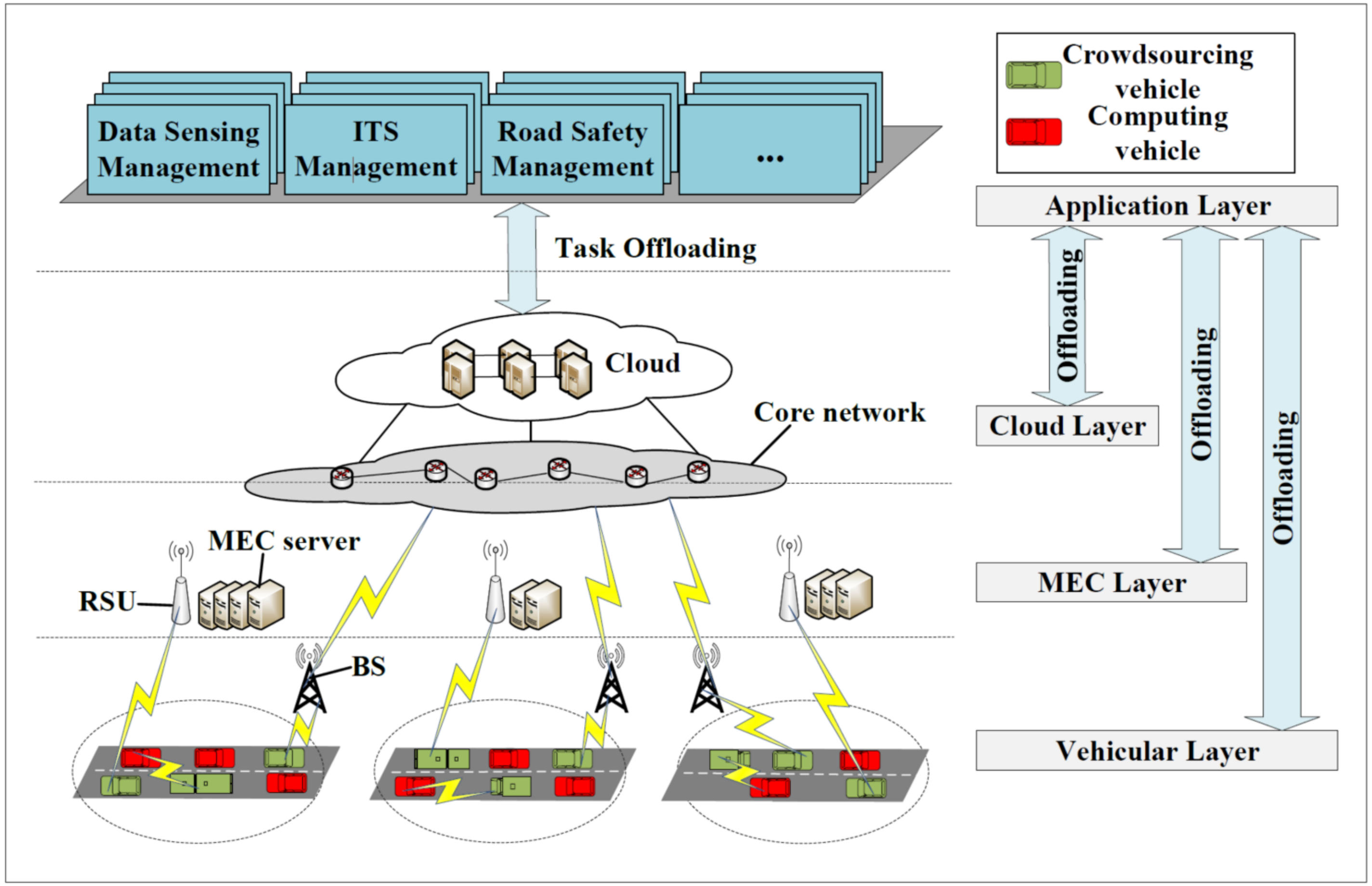


Fig. 5 Service Architecture

01 Task transmission model

Allocation ratio

- MEC: transmission time via V2I $tt_{r_v,m} = \frac{||D_{r_v}||}{\boxed{x_{r_v}} \cdot b_m \cdot log_2(1+\frac{P_v \cdot g_{mv}}{N_0})}$ Allow concurrent transmission with non-profit
- Vehicle: transmission time via V2V $tt_{r_v,v'} = \frac{||D_{r_v}||}{b_{v'} \cdot log_2(1+\frac{P_v g_{vv'}}{N_0})}$ One-to-one transmission with non-profit
- Cloud: transmission time via Cellular interface $tt_{r_v,c} = \frac{||D_{r_v}||}{b_c \cdot log_2(1+\frac{P_v \cdot g_{vc}}{N_0})}$ Anywhere transmission with cost

02 Task computation model

Allocation ratio

- Computation time at MEC server: $pt_{r_v,m} = \frac{cr_{r_v}}{\boxed{y_{r_v}} \cdot f_m}$ Limited concurrent transmission with non-profit
- Computation time at computing vehicle: $pt_{r_v,v'} = \frac{cr_{r_v}}{f_{v'}}$ At most one task at a time
- Computation time at cloud: $pt_{r_v,c} = \frac{cr_{r_v}}{\boxed{f^c_{r_v}}}$ Unlimited resources

Renting cost
$\omega_c > \omega_m > \omega_v$

03 Objectives

- Service time

$$st_{r_v} = \sum_{\forall l \in N_v} a^l_{r_v} (tt_{r_v,l} + pt_{r_v,l})$$

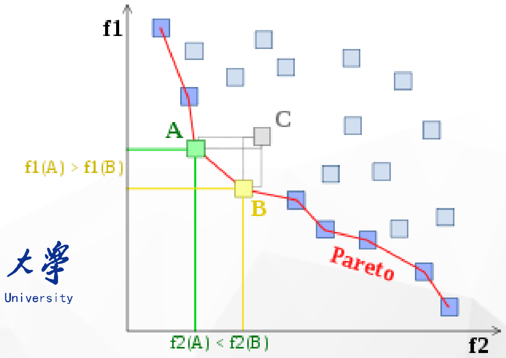
- Service cost

$$sc_{r_v} = a^c_{r_v} \cdot tc_{r_v,c} + \sum_{\forall l \in N_v} a^l_{r_v} \cdot pc_{r_v,l}$$

- Weighted sum

$$f = \eta_1 \cdot f_1 + \eta_2 \cdot f_2 = \sum_{\forall r \in R} \sum_{\forall r_v \in r} \frac{\eta_1 \cdot st_{r_v} + \eta_2 \cdot sc_{r_v}}{||r|| \cdot ||R||}$$

- Alternative: Pareto solution



https://en.wikipedia.org/wiki/Pareto_efficiency

04 Optimization model (mixed integer non-linear)

$$\min_{\boxed{\mathbf{A,X,Y,F}}} \quad f = \sum_{\forall r \in R} \frac{\eta_1 \cdot st_r + \eta_2 \cdot sc_r}{||R||}$$

$$\text{s.t.} \quad \sum_{\forall l \in N_v} a^l_{r_v} = 1, \forall r \in R, \forall v \in V^s$$

$$\sum_{\forall r \in R} \sum_{\forall r_v \in r} a^m_{r_v} x_{r_v} \leq 1, \forall m \in M$$

$$\sum_{\forall r \in R} \sum_{\forall r_v \in r} a^m_{r_v} y_{r_v} \leq t_m, \forall m \in M$$

$$a^l_{r_v} tt_{r_v,l} \leq L_{vl}, \forall l \in N_v$$

Non-linear

$$\sum_{\forall r \in R} \sum_{\forall v \in V^s_{v'}} a^{v'}_{r_v} \leq 1, \forall v' \in V^c_m$$

$$a^l_{rv} \in \{0,1\}, \forall l \in N_v, \forall r_v \in r, \forall r \in R$$

$$f^c_{r_v} \in R^+, \forall r_v \in r, \forall r \in R$$

Asynchronous Deep Q-Learning for Task Offloading

- Framework (A3C)[4]
  - ✓ Global model at cloud
  - ✓ Local agent at edge
  - ✓ Local training
  - ✓ Asynchronous update
- DQN model
  - ✓ $Q(s,a,\theta) \approx Q'(s,a)$
  - ✓ Target Q and Q network
  - ✓ Loss $L(\theta_m) = E[(Q_{target} - Q(s,a,\theta_m))^2]$
  - ✓ Parameter update $\theta_m = \theta_m + \beta \nabla_{\theta_m} L$

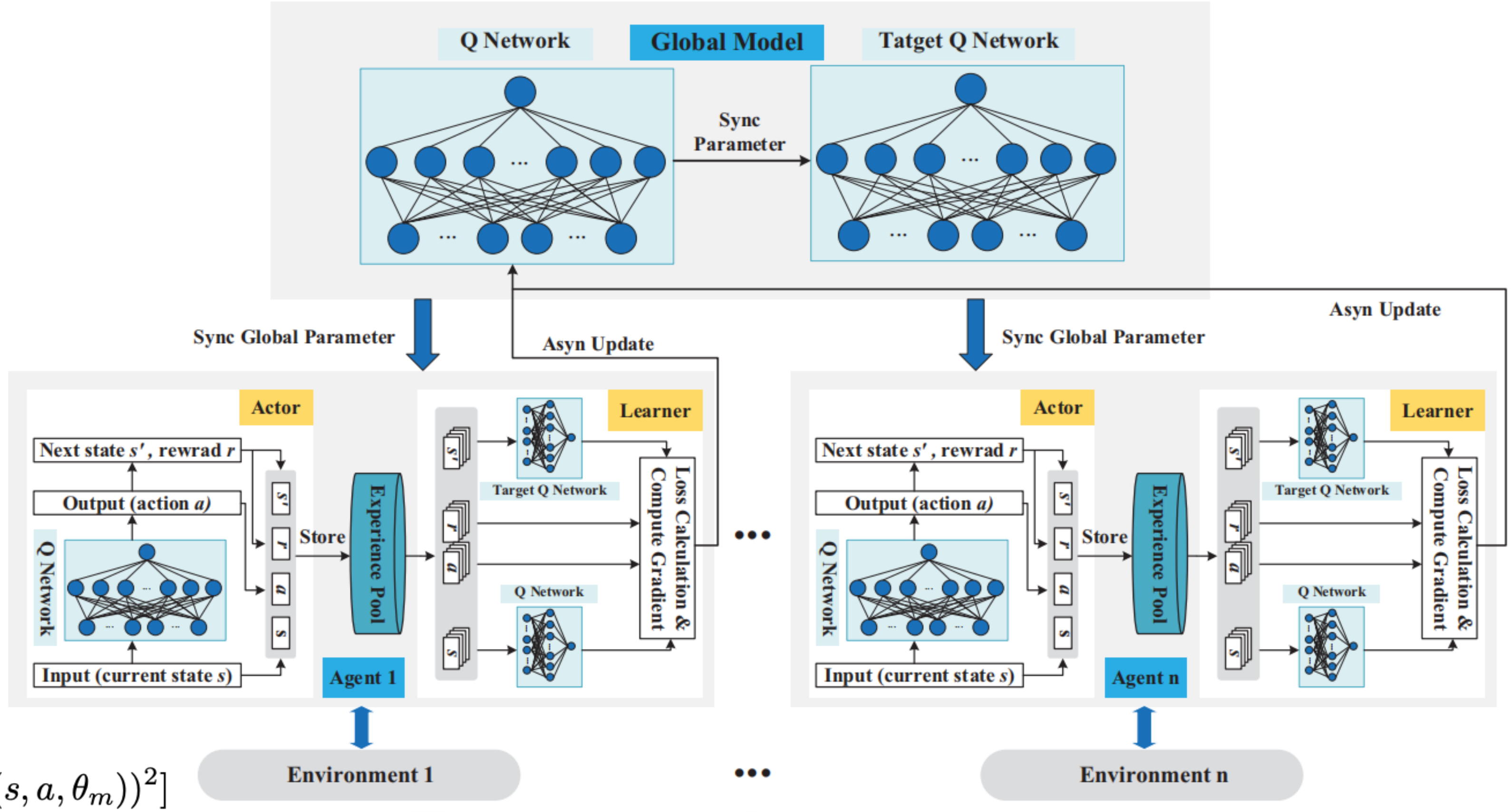


Fig. 6 The Diagram of Asynchronous Deep Q Learning Algorithm

[4] Mnih V, Badia A P, Mirza M, et al. Asynchronous methods for deep reinforcement learning[C]//International conference on machine learning. 2016: 1928-1937.

Asynchronous Deep Q-Learning for Task Offloading

- System state
  - ✓ at most n computing vehicles are available to each crowdsourcing vehicle

Pending task The real-time workload of m

$$s_{r_v}(t)=[D_{r_v},cr_{r_v},D_{total},D_{load},b_m,f_m,b_{v'_1},f_{v'_1},\cdots,b_{v'_n},f_{v'_n}]$$

- Action space: one-hot encoding

$$u_{r_v}(t)=[a_{r_v}^{m},a_{r_v}^{c},a_{r_v}^{v'_1},a_{r_v}^{v'_2},\cdots,a_{r_v}^{v'_n}]$$

- Reward function
  - ✓ Delay and cost

$$r_{r_v}(t)=\begin{cases}\dfrac{M_1}{\eta_1 st_{r_v}+\eta_2 sc_{r_v}}, & st_{r_v,l}-L_{v,l}<0\\ -M_2, & otherwise\end{cases}$$

Optimal Resource Allocation based on Convex optimization

$$\min_{X,Y,F} f = \sum_{\forall r\in R} \frac{\eta_1 \cdot st_r + \eta_2 \cdot sc_r}{\| R \|}$$

$$s.t \quad \sum_{\forall r\in R}\sum_{\forall v\in V_m^s} a_{r_v}^m x_{r_v} \le 1, \forall m \in M$$

$$\sum_{\forall r\in R}\sum_{\forall v\in V_m^s} a_{r_v}^m y_{r_v} \le t_m, \forall m \in M$$

$$f_{r_v}^c \in R^+, \forall r_v \in r, \forall r \in R$$

①cloud

$$\min_{f_{r_v}^c} \quad g_1 = \eta_1 \cdot \frac{cr_{r_v}}{f_{r_v}^c} + \eta_2 \cdot \omega_c \cdot f_{r_v}^c$$

$$s.t. \quad f_{r_v}^c \in R^+, \forall r_v \in r, \forall r \in R$$

$\nabla g_1 = 0$

$$f_{r_v}^{c*} = \sqrt{\frac{\eta_1 cr_{r_v}}{\eta_2 \omega_c}}, \forall r_v \in r, \forall r \in R$$

②Comm resource allocation of MEC

$$\min_{X} \quad g_2 = \sum_{\forall r\in R}\sum_{\forall r_v\in r} \frac{\eta_1 a_{r_v}^m tt_{r_v,m}}{\| r \| \cdot \| R \|}$$

$$s.t. \quad \sum_{\forall r\in R}\sum_{\forall v\in V_m^s} a_{r_v}^m x_{r_v} \le 1, \forall m \in M$$

decompose

$$\min_{X_m} \quad g_2^m = \sum_{\forall r\in R^m}\sum_{\forall r_v\in r} \frac{\eta_1 a_{r_v}^m tt_{r_v,m}}{\| r \| \cdot \| R \|}$$

$$s.t. \quad \sum_{\forall r_v\in R^m} a_{r_v}^m x_{r_v} \le 1$$

KKT

$$x_{r_v}^* = \frac{a_{r_v}^m \sqrt{\xi_{r_v}}}{\sum_{\forall r_v\in R^m} a_{r_v}^m \sqrt{\xi_{r_v}}}, \forall r_v \in R^m$$

$$\xi_{r_v} = \frac{\eta_1 \cdot D_{r_v}}{b_m \cdot \| r \| \cdot \log_2 (1 + \frac{P_v g_{mv}}{N_0})}, \forall r_v \in R^m$$

③Comp resource allocation of MEC

$$\min_{Y} \quad g_3 = \sum_{\forall r\in R}\sum_{\forall r_v\in r} \frac{\eta_1 a_{r_v}^m pt_{r_v,m} + \eta_2 a_{r_v}^m pc_{r_v,m}}{\| r \| \cdot \| R \|}$$

$$s.t. \quad \sum_{\forall r\in R}\sum_{\forall v\in V_m^s} a_{r_v}^m y_{r_v} \le t_m, \forall m \in M$$

decompose

$$\min_{Y_m} \quad g_3^m = \sum_{\forall r_v\in R^+} \frac{\eta_1 a_{r_v}^m pt_{r_v,m} + \eta_2 a_{r_v}^m pc_{r_v,m}}{\| r \| \cdot \| R \|}$$

$$s.t. \quad \sum_{\forall r_v\in R^m} a_{r_v}^m y_{r_v} \le t_m$$

KKT1: $\lambda_m = 0$

$$y_{r_v}^* = a_{r_v}^m \sqrt{\frac{\theta_{r_v}}{\gamma_{r_v}}}, \forall r_v \in R^m$$

$$\theta_{r_v} = \frac{\eta_1 \cdot cr_{r_v}}{f_m \cdot \| r \| \cdot \| R \|}, \forall r_v \in R^m$$

$$\gamma_{r_v} = \frac{\eta_2 \cdot \omega_m \cdot f_m}{\| r \| \cdot \| R \|}, \forall r_v \in R^m$$

KKT2: $\lambda_m \neq 0$

$$y_{r_v}^* = a_{r_v}^m \sqrt{\frac{\theta_{r_v}}{\gamma_{r_v} + \lambda_m}}, \forall r_v \in R^m$$

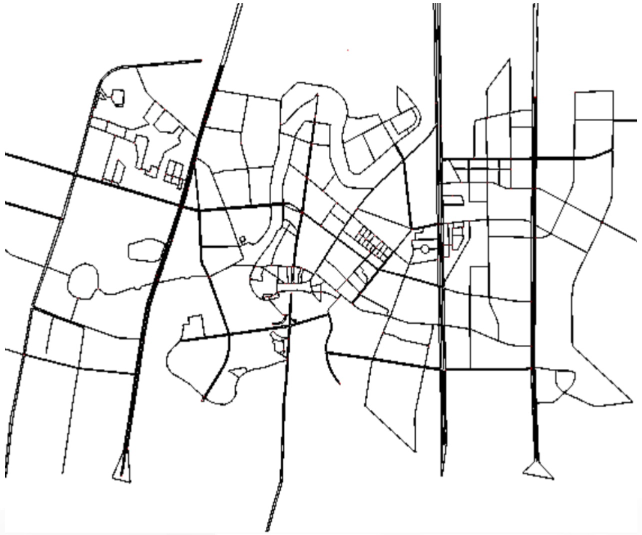
$$\sum_{\forall r_v\in R^m} a_{r_v}^m \sqrt{\frac{\theta_{r_v}}{\gamma_{r_v} + \lambda_m}} - t_m = 0$$

Default setting

SUMO Environment

1. 4km × 4km area of Tianfu new district in Chengdu, China

2. Five MEC servers distributed among road network



Simulation parameters

Parameter	Value
Data size of a task	[45, 75] MB
Required computing resources	[30, 50] G CPU
Wireless bandwidth of vehicle/MEC	30/150 MHz
Comp capacity of vehicle/MEC	10/60 G CPU
The unit price of comp resource of vehicle/MEC/cloud	0.05/0.2/1 $/G cycles

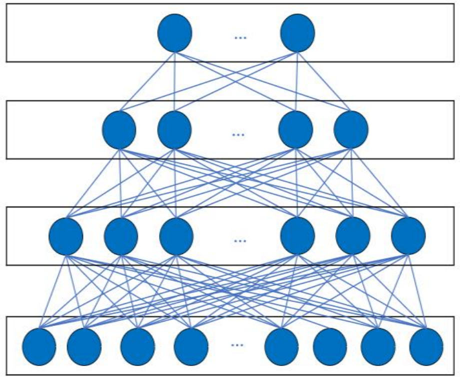
Metrics

1. Average service time
2. Average service cost
3. Objective $f = \eta_1 \cdot f_1 + \eta_2 \cdot f_2$

Network parameters

1. The learning rate and discount factor is set to 0.001 and 0.9

2. The size of experience pool and batch is set to 100 and 10

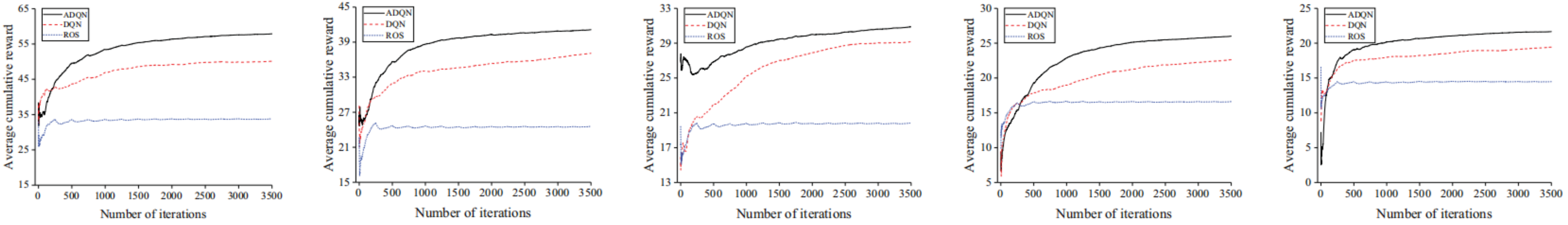


Comparison algorithm

1. Deep Q-learning (DQN)
2. Random offload scheduling(ROS)

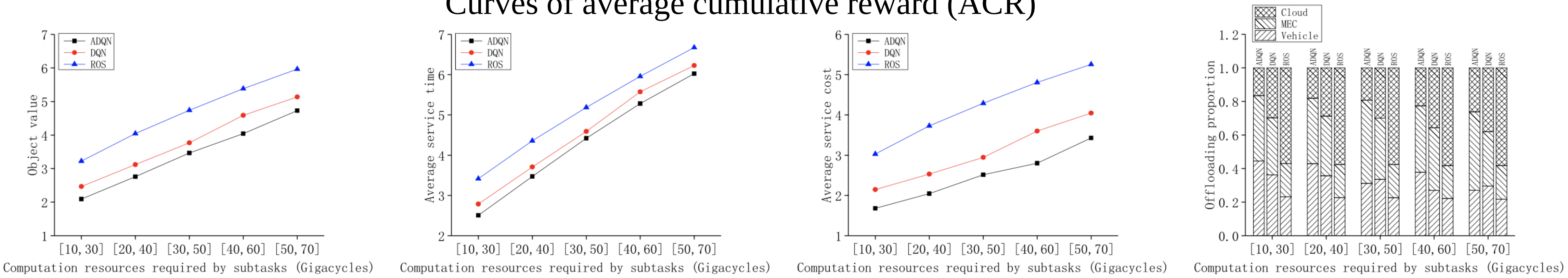
Simulation result

1. Effect of computation resource requirement



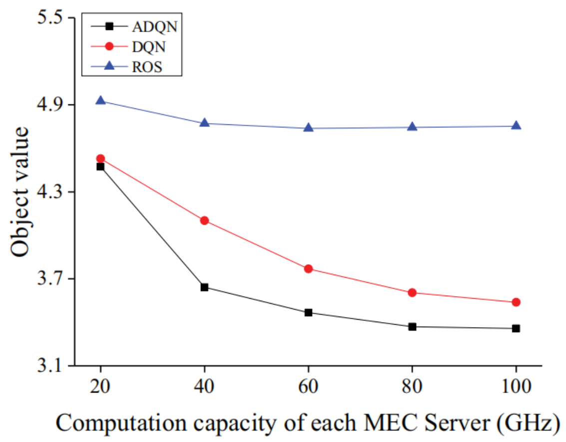
(a) cr_{rv} ~[10,30] (b) cr_{rv} ~[20,40] (c) cr_{rv} ~[30,50] (d) cr_{rv} ~[40,60] (e) cr_{rv} ~[50,70]

Curves of average cumulative reward (ACR)

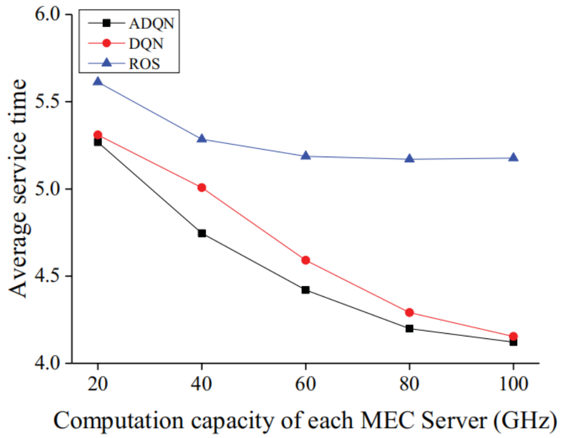


(a) OV (b) AST (c) ASC (d) OP

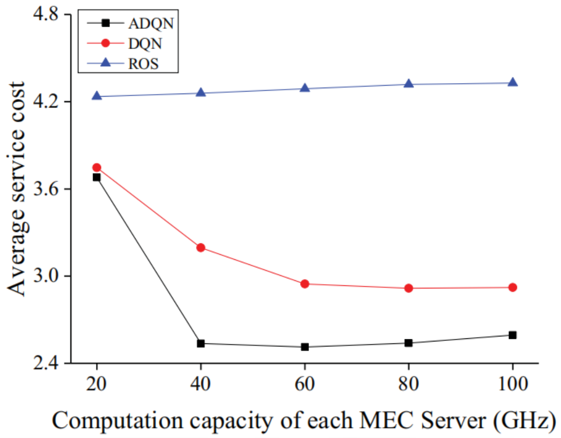
2. Effect of computation capacity of MEC server



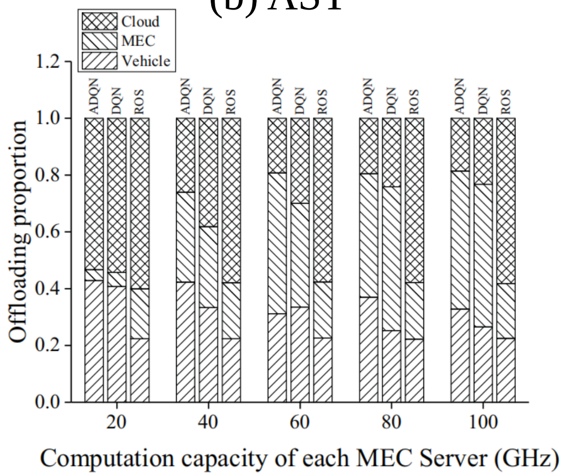
(a) OV



(b) AST



(c) ASC



(d) OP

THANK YOU

Email:albertdai094@gmail.com