Deep Q-Learning based Joint Dynamic Network Slicing and Resource Allocation in Multi-Tenant Edge Computing System

Network slice parameter α, β

Task format:

$J_i = (\rho_i, a_i, d_i, T_i^*)$

Local execution time:

$$T_i^l = \frac{(1 - x_i)d_i}{F_i} \tag{1}$$

Data rate:

$$r_i = b_i \log_2(1 + \frac{g_i p_i}{\sigma^2}) \tag{2}$$

Transmission time:

$$T_i^{tx} = \frac{x_i a_i}{r_i} \tag{3}$$

Execution time:

$$T_i^{ex} = \frac{x_i d_i}{f_i} \tag{4}$$

Remote execution time:

$$T_i^o = T_i^{tx} + T_i^{ex} \tag{5}$$

Utility:

$$U_i(x, b, f) = \begin{cases} \Psi \quad \text{if } T_i > T_i^*, \\ \max(T_i^o, T_i^l) \qquad \text{otherwise.} \end{cases} \tag{6}$$

Revenue:

$$Rev_z = \sum_i \rho_i U_i \tag{7}$$

Cost:

$$C_z = \zeta\alpha + \xi\beta \tag{8}$$

Problem formulation:

$$\begin{aligned} &\max \frac{1}{n} \sum_{z=0}^{n} Rev_z - Cost_z \\ \text{s.t.} \quad & C1 : 1 \\ & C2 : 2 \end{aligned} \tag{9}$$

SP1:

$$\begin{aligned} &\max_{\alpha,\beta} E[Rev_z(\alpha, \beta) - Cost_z(\alpha, \beta)] \\ \text{s.t.} \quad & C1 : \alpha \in [0, 1] \\ & C2 : \beta \in [0, 1] \end{aligned} \tag{10}$$

RL:

$$r_{s,a} = \begin{cases} \Phi, \quad \text{if } \sum_{n \in F} \sum f_{n,w} > F, \\ \sum_{i=0}^{n} \varphi C_i(x_i, f_i, w_i, e_i) \quad\quad \text{otherwise.} \end{cases} \tag{11}$$

$$V^{\pi} = E[\Sigma_{i=0}^{\inf} \gamma r_i | s_0 = s] \tag{12}$$

$$\pi^* = arg \max_{\pi} Q(s,a), \forall s \in S \tag{13}$$

$$Q^*(s,a) = Q(s,a) + \alpha(r + \beta \max Q(s,a) - Q(s,a)) \tag{14}$$

$$Loss(\theta) = E[y - Q(s,a,\theta)^2] \tag{15}$$

SP2:

$$\begin{aligned} &\max_{x,b,f} \sum_i \rho_i U_i(x_i, b_i, f_i) \\ \text{s.t.} \quad & C1 : x_i \in [0,1] \\ & C2 : b_i \in [0,B] \\ & C3 : f_i \in [0,F] \\ & C4 : \sum_i b_i = B \\ & C5 : \sum_i f_i = F \end{aligned} \tag{16}$$

Theorem 1: When minimum occur,

$$x_i = \frac{d_i}{F_i(\frac{a_i}{b_i \log_2(1+\frac{g_i p_i}{\sigma^2})} + \frac{d_i}{f_i} + \frac{d_i}{F_i})} \tag{17}$$

Proof: For i

$$U_i = max(T_i^o, T_i^l) = max(x_i(\frac{a_i}{b_i \log_2(1+\frac{g_i p_i}{\sigma^2})} + \frac{d_i}{f_i}), \frac{d_i}{F_i}) \tag{18}$$

$$T_i^o = T_i^l, x_i(\frac{a_i}{b_i \log_2(1+\frac{g_i p_i}{\sigma^2})} + \frac{d_i}{f_i}) = \frac{(1-x_i)d_i}{F_i} \tag{19}$$

After rearange equation (), we can obtain $x_i = \frac{d_i}{F_i(\frac{a_i}{b_i \log_2(1+\frac{g_i p_i}{\sigma^2})} + \frac{d_i}{f_i} + \frac{d_i}{F_i})}$.

Reformulate:
relax U_i to $\hat{U}_i = ...$ no penalty

$$\max_{x,b,f} \sum_i \rho_i x_i (\frac{a_i}{b_i \log_2{(1+\frac{g_i p_i}{\sigma^2})}} + \frac{d_i}{f_i})$$
$$\text{s.t.} \quad C2 - C5 \tag{20}$$

Proposition 2: Objective is convex

Proof: The Hessian matrix of (12) consists of, where. Thus, the Hessian matrix of (12) is a positive definite matrix, and hence is convex.

$$\frac{\partial^2}{\partial^2 b_i^2} = ... >= 0 \tag{21}$$

$$\frac{\partial^2}{\partial^2 f_i^2} = ... >= 0 \tag{22}$$

Theorem 2: Given a set of offloading request, the optimal bandwidth and computing resource allocation is

$$b_i = \frac{\sqrt{\frac{\rho_i x_i a_i}{\log_2{(1+\frac{g_i p_i}{\sigma^2})}}}}{\sum_i \sqrt{\frac{\rho_i x_i a_i}{\log_2{(1+\frac{g_i p_i}{\sigma^2})}}}} B, f_i = \frac{\sqrt{\rho_i x_i d_i}}{\sum_i \sqrt{\rho_i x_i d_i}} F \tag{23}$$

Proof: We introduce Lagrange multiplier to solve the problem (12). The Lagrange dual function of (12) is

$$L = \sum_i \rho_i x_i (\frac{a_i}{b_i \log_2{(1+\frac{g_i p_i}{\sigma^2})}} + \frac{d_i}{f_i}) + \lambda(\sum_i b_i - B) + \mu(\sum_i f_i - F) \tag{24}$$

$$\begin{cases} \frac{\partial L}{\partial b_i} = +\lambda = 0, \forall i \\ \frac{\partial L}{\partial \lambda} = \sum_i b_i - B = 0 \end{cases} \tag{25}$$

$$\begin{cases} \frac{\partial L}{\partial f_i} = -\frac{\rho_i x_i d_i}{f_i^2} + \mu = 0, \forall i \\ \frac{\partial L}{\partial \mu} = \sum_i f_i - F = 0 \end{cases} \tag{26}$$

After solve equation (), we can obtain $b_i = \frac{\sqrt{\frac{\rho_i x_i a_i}{\log_2{(1+\frac{g_i p_i}{\sigma^2})}}}}{\sum_i \sqrt{\frac{\rho_i x_i a_i}{\log_2{(1+\frac{g_i p_i}{\sigma^2})}}}} B, f_i = \frac{\sqrt{\rho_i x_i d_i}}{\sum_i \sqrt{\rho_i x_i d_i}} F$.

Alg:

init:$x_i = min(1, \frac{T_i^* F_i^l}{d_i}), \forall i$

Algorithm 1 QoE-Driven Iterative Video Adaptation

Input: The set of

Initialization: replay memory $D = \emptyset$, policy network Q, target network Q

for episode = 1, M **do**
    Initialize state
    while current state is not a terminal state **do**
        Randomly select an action with probability ε
        Otherwise, select the action $a = maxQ$ according to policy network Q
        Execute the action observe reward r
        Store transition (s_i, s_{i+1}, a, r) in D
        Sample random a minibatch of transitions from D
        $y =$
        Update weights θ of policy network Q via a learning step with optimizer
        Reset target network Q $\leftarrow$Q every C steps
    end while
end for

Output: The set of