IEEE GlobalSIP 2018

Joint Content Popularity Prediction and Content Delivery Policy for Cache-Enabled D2D Networks :A Deep Reinforcement Learning Approach

Jiaying Yin[1], Lixin Li[1], Yang Xu[1], Wei Liang[1] , Huisheng Zhang[1], and Zhu Han[2]

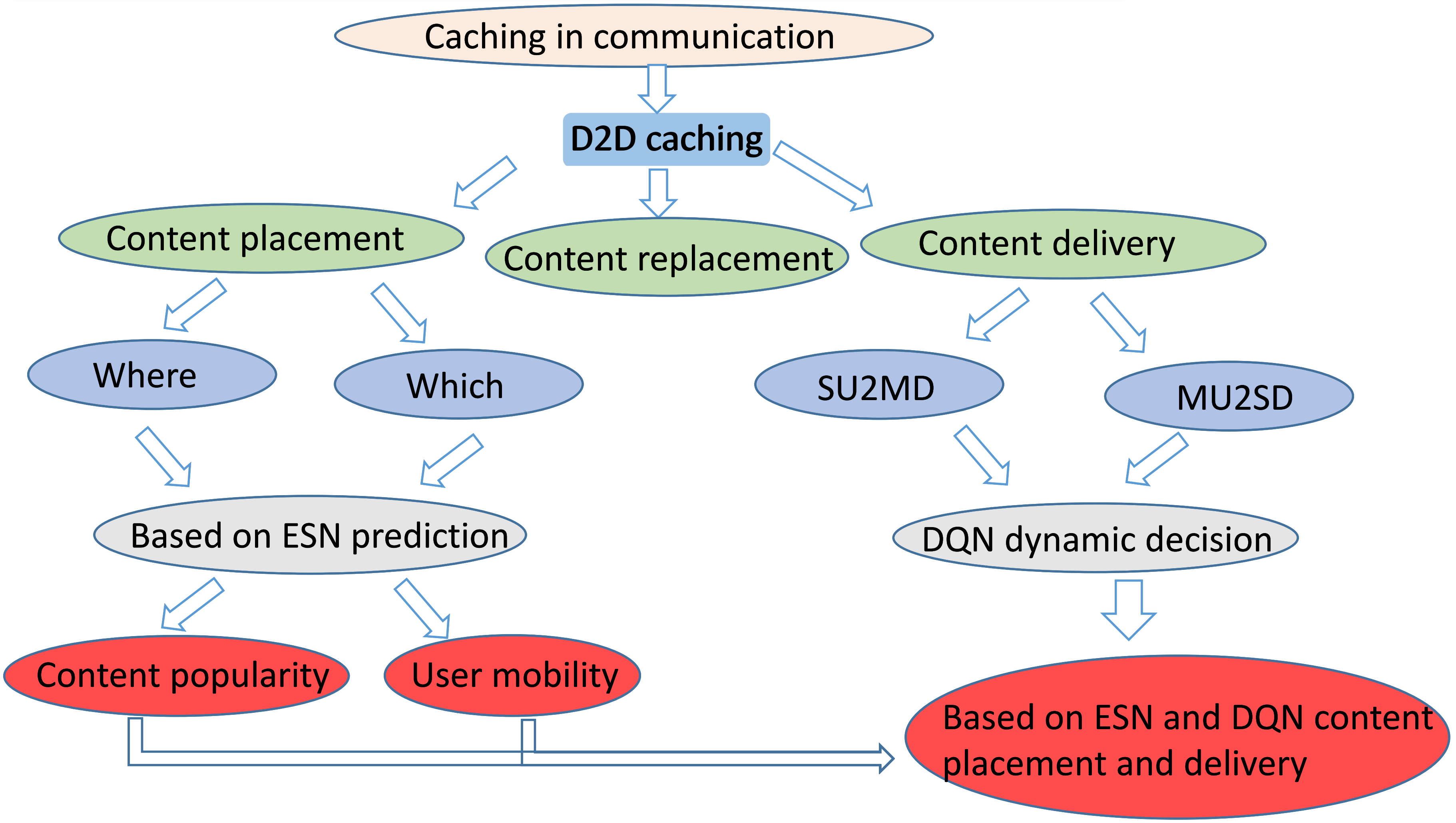
[1]School of Electronics and Information, Northwestern Polytechnical University, Xi'an, CHINA
[2]Department of Electrical and Computer Engineering, University of Houston, Houston, TX, USA

2018 IEEE Global Conference on Signal and Information Processing
Anaheim, California • November 26-28, 2018

Nov. 29, 2018

Content

Introduction



Introduction

Contributions in this paper:

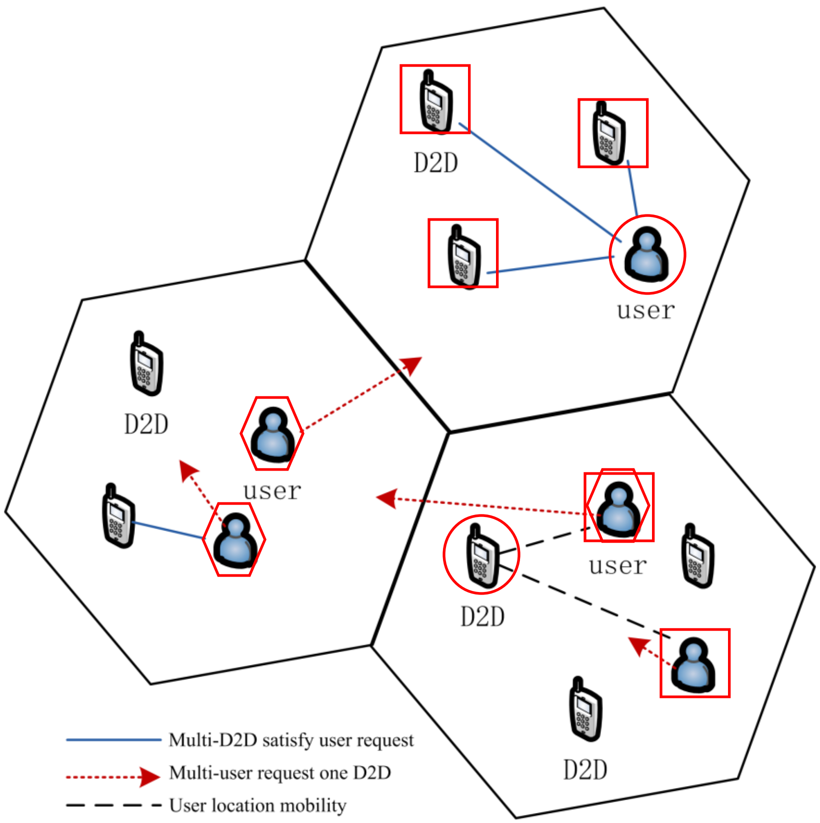
- We propose an ESN-based algorithm to predict both the content popularity and user mobility , thus determining which content to cache and where to cache.

- A DQN-based dynamic decision optimization for request content delivery is proposed with the channel state information and content transmission delays regarded as criteria .

- We formulate a reward function by adjusting the weight coefficients to tradeoff the overall optimization goals, and simulating the performance from the perspective of D2D device and user, respectively.

System model

The optimization goal is to maximize the cache hit rate(CHR) and reduce the overall system's transmission delay and the transmission power consumption.



$$\max \sum_{k=1}^{\mathcal{K}} c_k L$$

$$\min \left\{ \xi \sum_{t=t_1}^{\mathcal{T}} T_{k,n}(t) + \eta \sum_{t=t_1}^{\mathcal{T}} p_{k,n}(t) \right\}$$

$$s.t. \quad c_k \in \{0,1\}, \forall n \in (1,N), \forall k \in \mathcal{K}$$

The transmission delay

The transmission power consumption

Algorithm implementation

Popularity Prediction

The state of input at time *t*

The number of input layer node

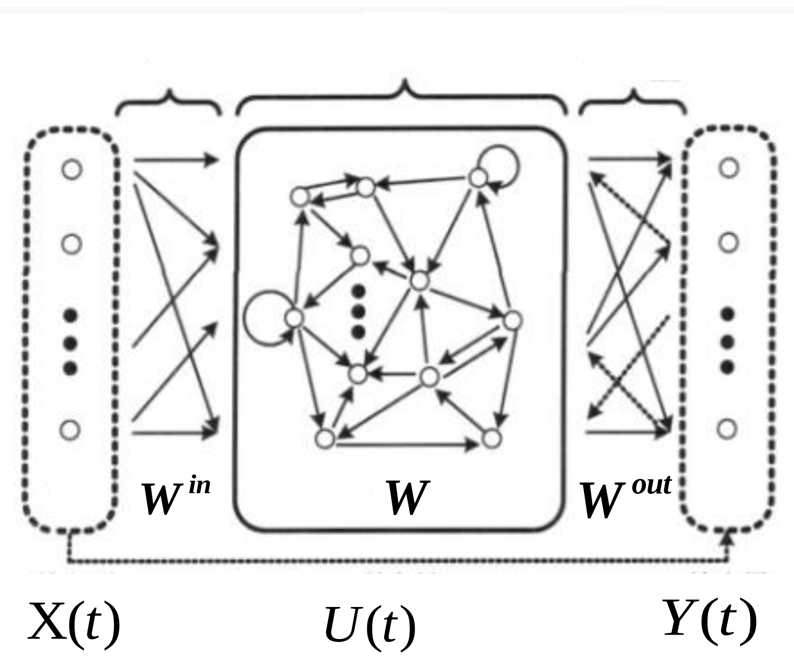
$$X(t) = [x_1(t), x_2(t), \cdots, x_K(t)],$$

$$U(t) = [u_1(t), u_2(t), \cdots, u_M(t)],$$

$$Y(t) = [y_1(t), y_2(t), \cdots, y_N(t)],$$

The number of output layer node

The state of output at time *t*



Algorithm implementation

The update of the hidden layer state and the output layer state of ESN at time *t+1* can be expressed as :

input layer matrix

Hidden layer matrix

the output layer of the previous moment to the hidden layer of the next moment

$$U(t+1)=f(\mathbf{W}^{in}X(t+1)+\mathbf{W}U(t)+\mathbf{W}^{back}Y(t)),$$

$$Y(t+1)=f_{out}(\mathbf{W}^{out}[X(t+1);U(t+1)]),$$

The activation function of output layer neurons

output layer matrix

The concatenation of two vectors

The real content request distribution of user

The prediction results of content request distribution

$$\min\{\sum_{k=m}^{P}(Y(k)-\sum_{i=1}^{N}f_{out}(\mathbf{W}_j^{out}[X(t);U(t)]))\},$$

$$\mathbf{W}^{out}=\mathbf{Y}\mathbf{U}^T(\mathbf{U}\mathbf{U}^T+1\mathbf{I})^{-1}$$

Algorithm implementation

Based on ESN to predict the content popularity and user's mobility

Content popularity

$$X_{tk} = [x_{t1}, x_{t2}, \cdots, x_{tK}]^T$$

The prediction results of content request distribution

$$Y_{tk} = [p_{tk1}, p_{tk2}, \cdots, p_{tkN}]^T$$

The real content request distribution of user

User's mobility

$$X_k = [l_{t-1,k}, l_{t-2,k}, \cdots, l_{t-L,k}]^T$$

The prediction results of content request distribution

$$r_{t,k} = [r_{tk1}, r_{tk2}, \cdots, r_{tkN'}]^T$$

The prediction results of content request distribution

Algorithm implementation

The Establishment of ESN

The network state set

$$S^t = (P_{k,n}^t, g_{k,n}^t, d_{k,n}^t) \in S = \{P_{k,1}, P_{k,2}, \cdots P_{k,\mathcal{N}}, g_{k,1}, g_{k,2}, \cdots g_{k,\mathcal{N}}, d_{k,1}, d_{k,2}, \cdots, d_{k,\mathcal{N}}\}$$

$$A^t = (u_{k,n}^t) \in A = \{u_{k,1}, u_{k,2}, \cdots, u_{k,\mathcal{N}}\}$$

The network action set

$$R_\pi^t(s,a) = (\xi \frac{-d_{k,n}^t}{\log_2(1 + \frac{P_{k,n}^t g_{k,n}^t d_{k,n}^{-\beta t}}{\sum_{k' \neq k} P_{k',n}^t g_{k',n}^t d_{k',n}^{-\beta t} + \delta^2})} - \eta \frac{d_{k,n}^t P}{g_{k,n}^t d_{k,n}^{-\beta t}})$$

Reward function

Reward function

$$V(s,\pi) = \sum_{t=1}^{\infty} (\gamma)^{t-1} R_\pi^t(s,a)$$

Reward function

$$\pi^* = \arg\max_\pi V(s,\pi), \forall s \in S$$

Performance

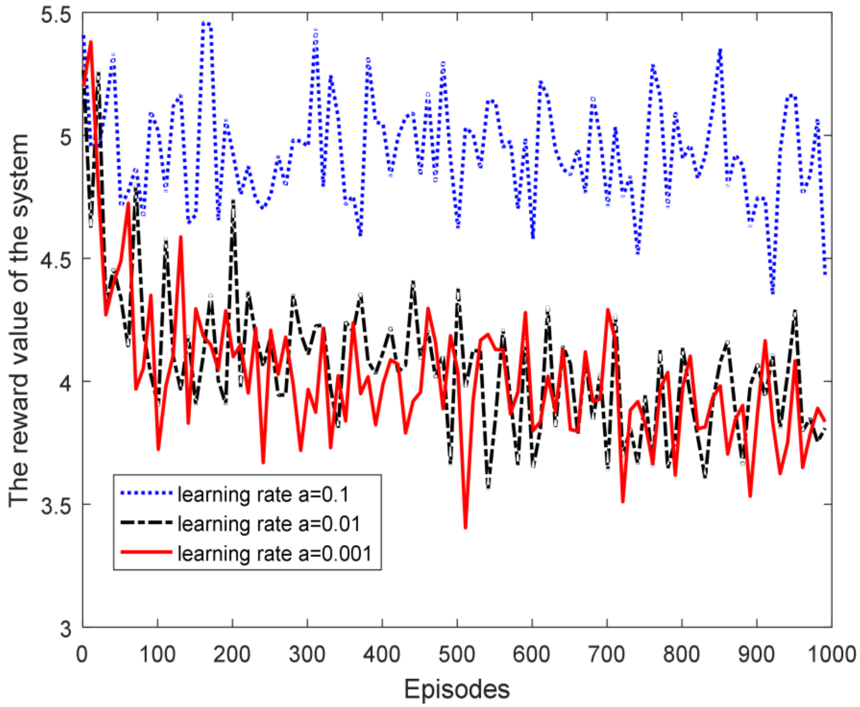


Fig. 1. Convergence performance of DQN-based algorithm under different learning rates

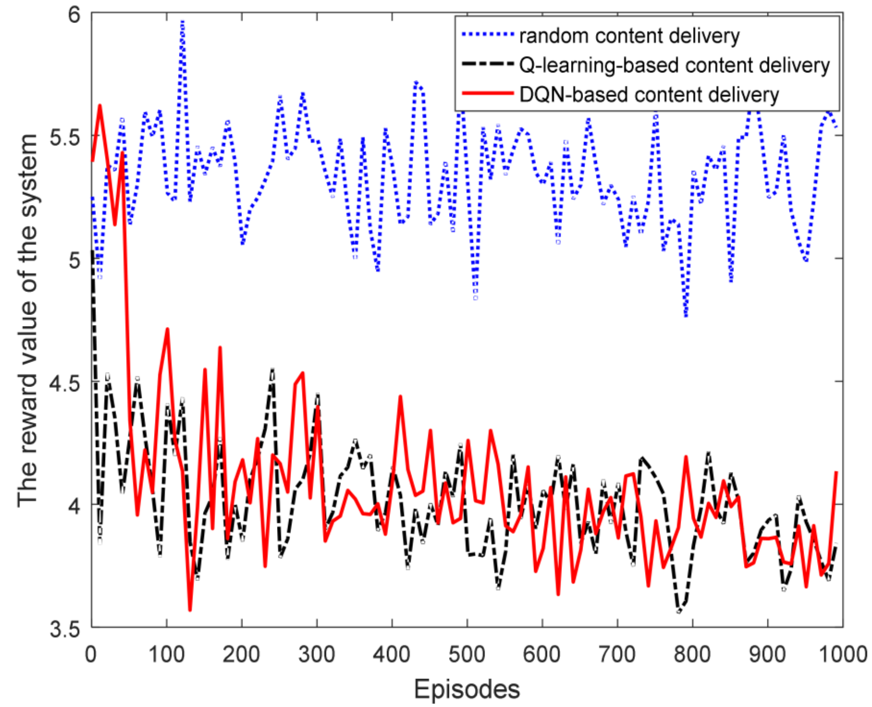


Fig. 2. The delivery costs with different delivery policies

Conclusions

From our studies and simulation results, we can have the following observations.

- CHR can be improved by selecting cache contents and cache location based on the ESN's prediction results of content popularity and user's mobility.
- DQN-based algorithm for dynamic decision-making of content delivery can decrease the delay and power consumption.
- Using different caching strategies for SBS to select caching content can improve the cache hit rate of the CCN.

Thank You

Contact: 18717382535@mail.nwpu.edu.cn