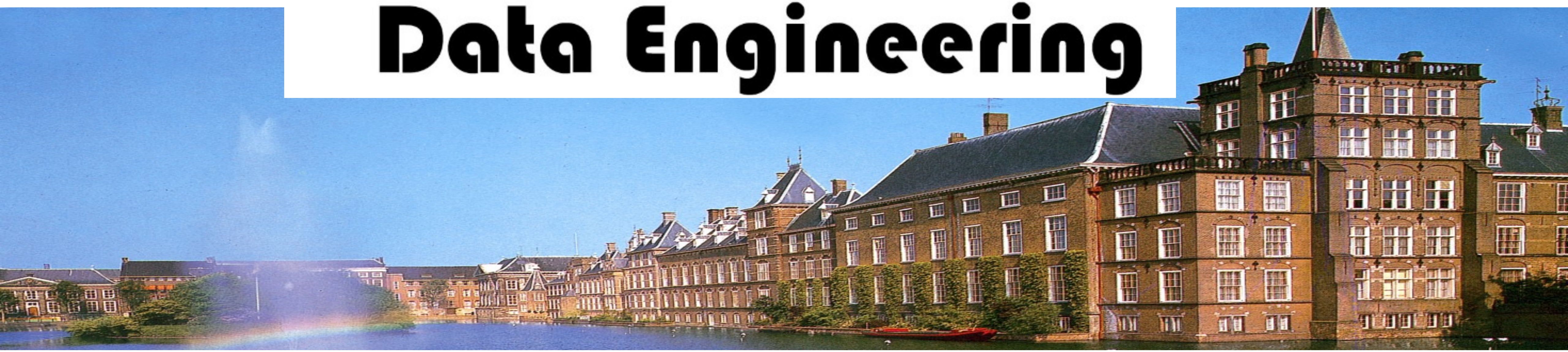
40th IEEE International Conference on



Joint Mobile Edge Caching and Pricing: A Mean-Field Game Approach

Yin Xu[1], Xichong Zhang[1] , Mingjun Xiao[1], Jie Wu[2], An Liu[3], and Sheng Zhang[4]

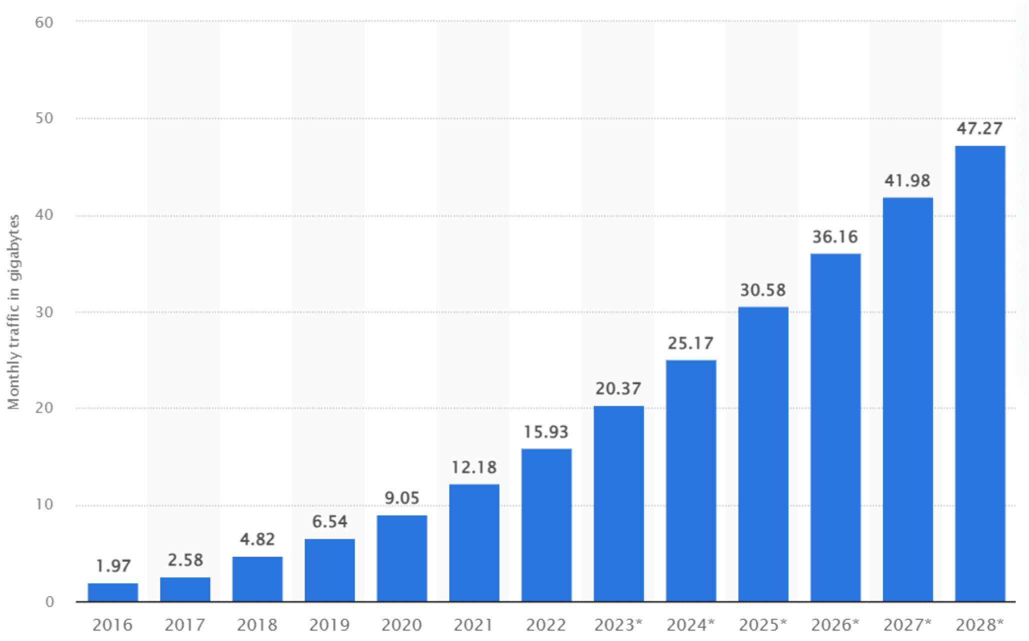
[1]University of Science and Technology of China
[2]Temple University [3]Soochow University [4]Nanjing University

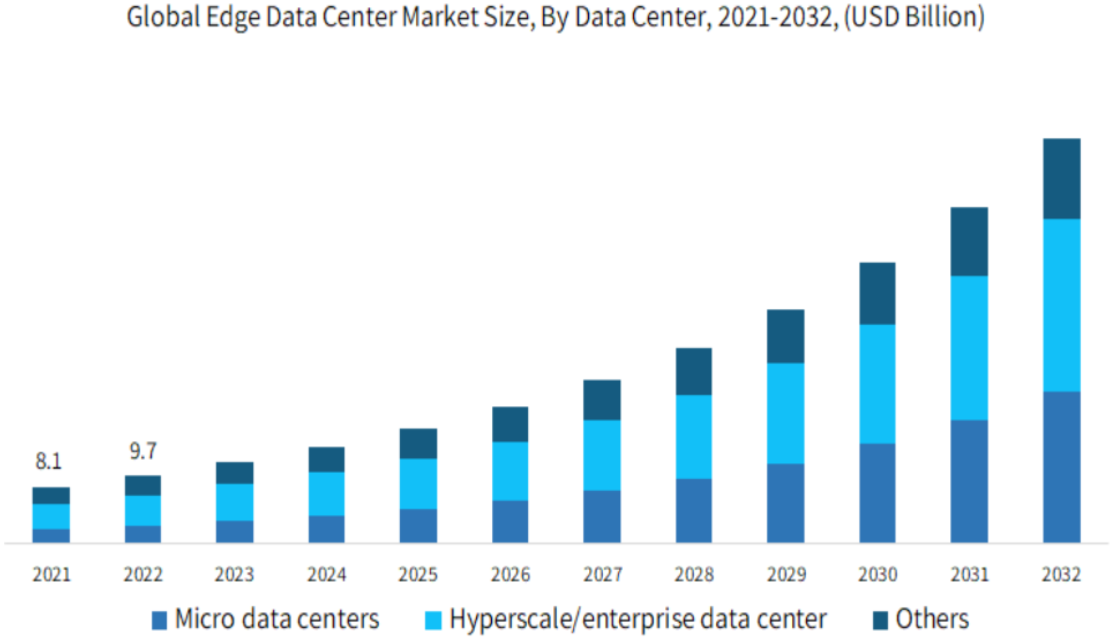
Outline

- Background & Motivation
- Related Work & Problem Formulation
- Basic Idea & Solution
- Evaluation & Conclusion

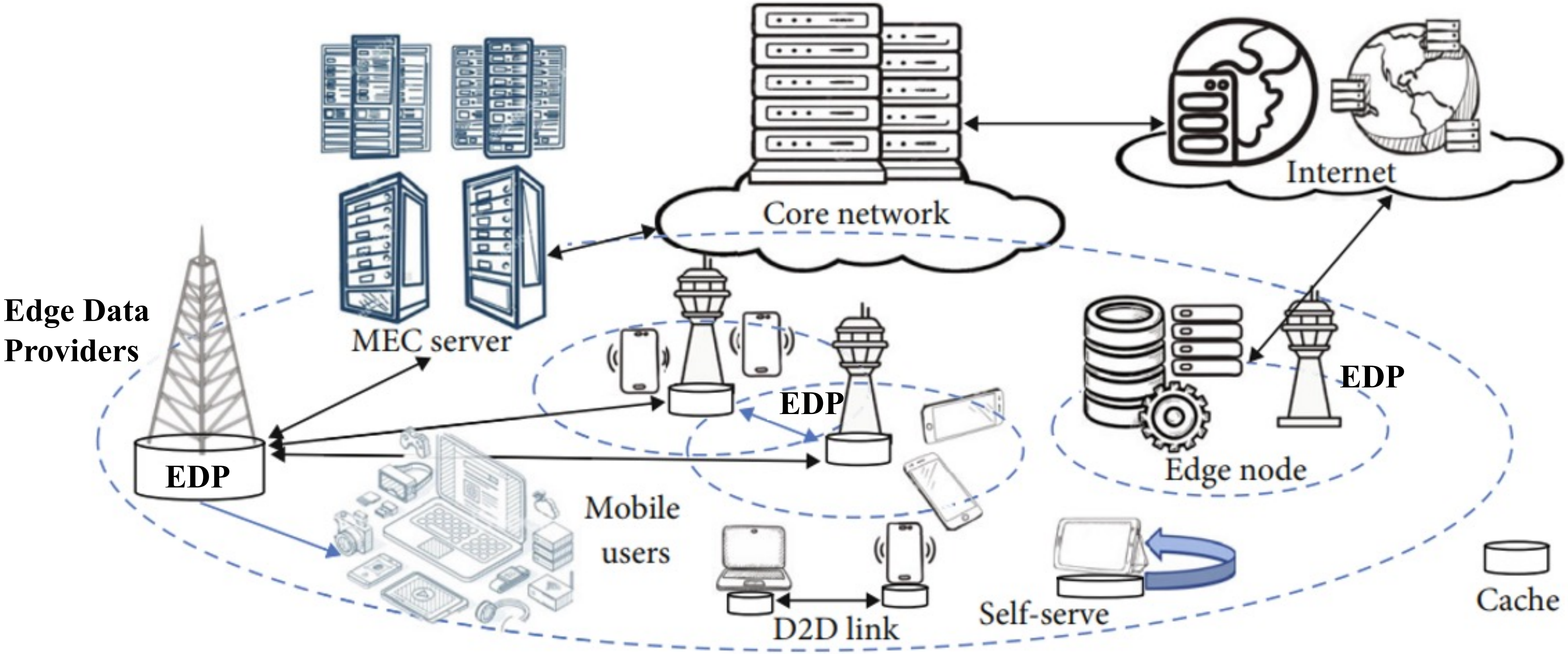
Background

- Explosive growth of mobile data traffic
  - Fast deployment of edge devices
  - 10x increase from 2016 to 2023
  - Significant stress on back-haul links
- Content caching and trading in the edge
  - By 2032, the market size of edge data centers is expected to exceed $50 billion
  - Enhance Quality of Experience (QoE)





Background

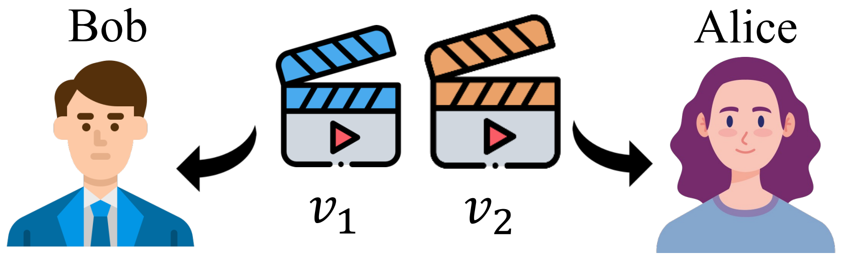


General architecture of Mobile Edge Caching systems

Motivation

Example

competitive content placement



Popularity: $v_1 > v_2$

- Both cache v_1 → A competition
  → Attract more requesters by sequentially lowering their trading prices
  → Low utilities or even negative utilities
  → Caching v_2 may be better
- Peer content sharing
  → Improve the caching efficiency

Economic effect of caching & sharing

Key Challenges

■ **Complicated interplay:**

■ **Real-time decision-making:**

- Dynamic network conditions
- Time-varying content service requests
  → popularity & timeliness demands

■ **Large-scale best response:**

- Non-cooperative game
- Distributed decisions
- Heavy overhead

Outline

- Background & Motivation
- Related Work & Game Formulation
- Basic Idea & Solution
- Evaluation & Conclusion

Related Work

Goal: edge caching optimization while considering pricing and content sharing

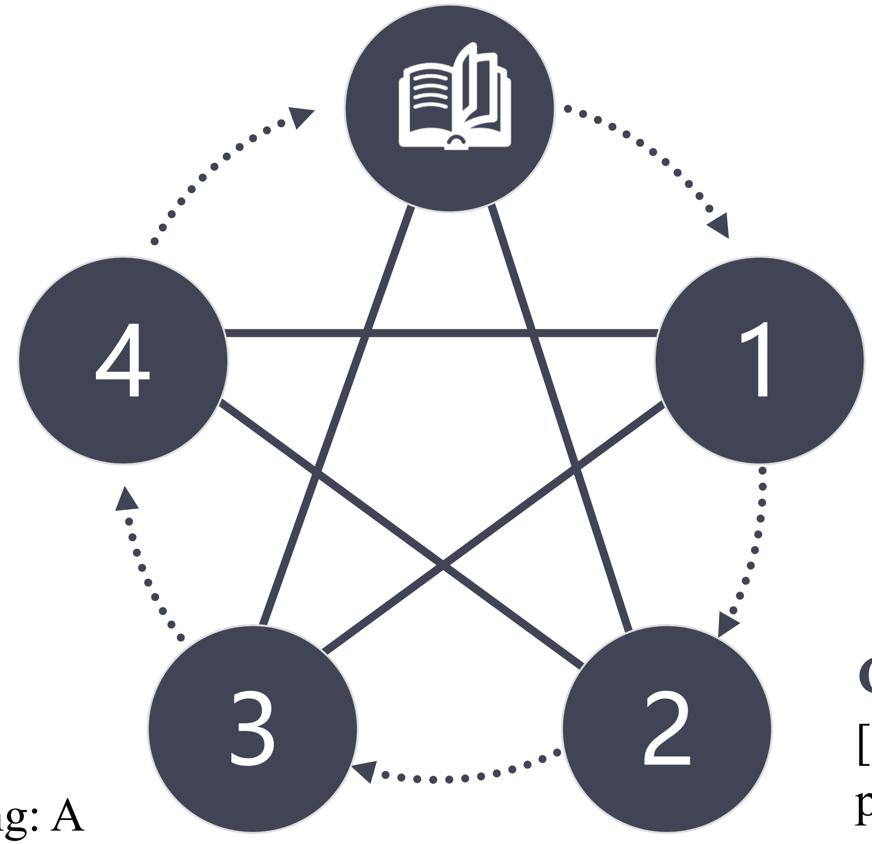
Mean-Field Game

[5] D. Narasimha, et al., “Age-dependent distributed MAC for ultra-dense wireless networks,” IEEE INFOCOM2021

[6] H. Feng, et al, “Mean-field game theory based optimal caching control in mobile edge computing,” IEEE TMC2022.

Edge caching and pricing

[4] J. Zou, et al, “ Joint pricing and cache placement for video caching: A game theoretic approach,” IEEE JSAC2019.



Mobile Edge Caching

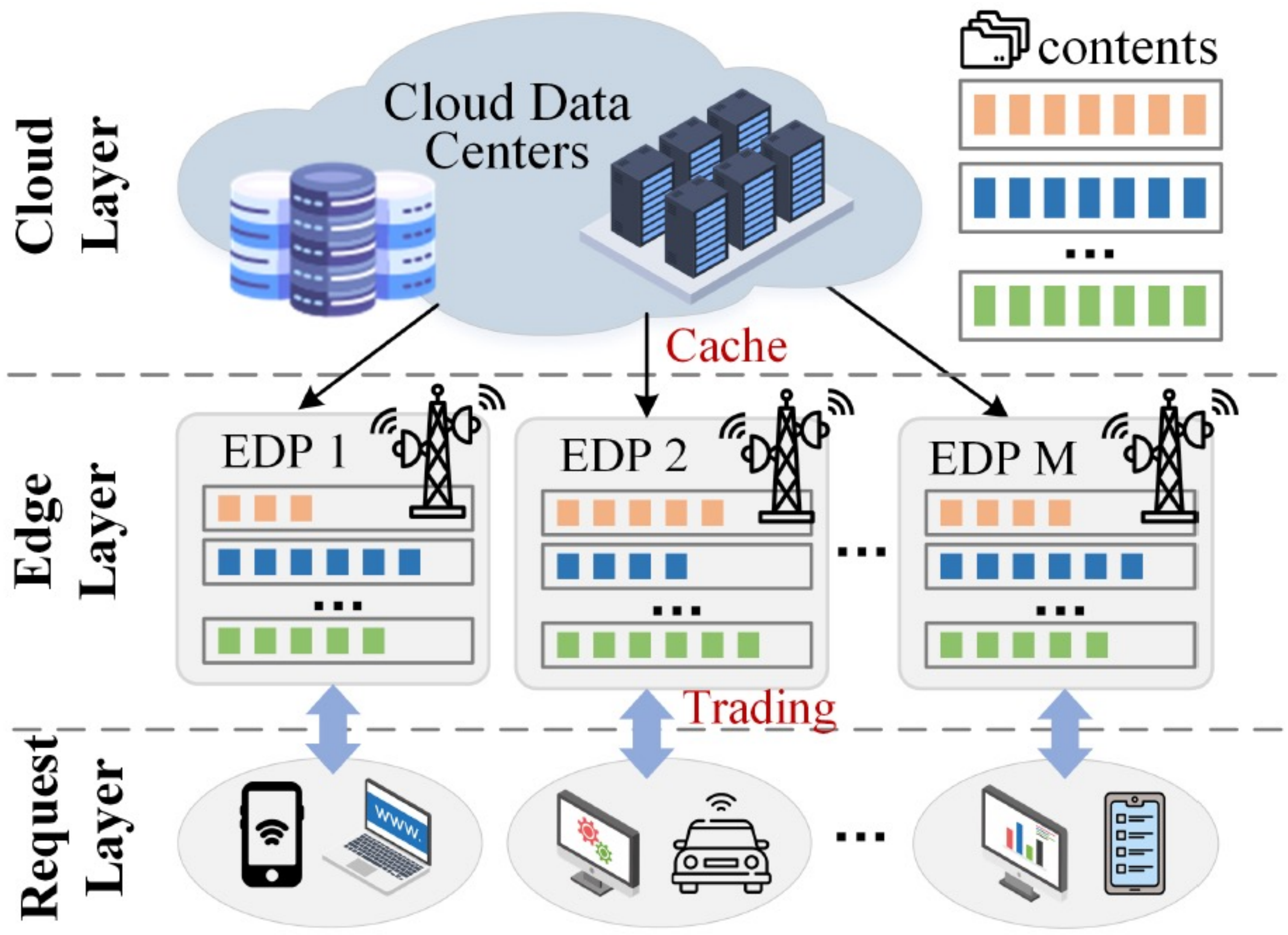
[1] H. Sun, et al, “ A proactive on-demand content placement strategy in edge intelligent gateways,” TPDS2023.

[2] T. Zong, et al, “ Cocktail edge caching: Ride dynamic trends of content popularity with ensemble learning ,” IEEE/ACM ToN2023.

Content Trading

[3] Y. Huang, et al, “ Profit sharing for data producer and intermediate parties in data trading over pervasive edge computing environments,” IEEE TMC2023.

System Model



System Model

- **Edge Data Providers (EDPs) {1,…,i,...,M}:** cache certain contents in advance and are allowed to share the cached contents with each other.
- **Content requesters {1,…,j,...,J}:** buy desired contents from EDPs and pay a suitable trading price.
- **Contents {1,…,k,...,K}:** the data size of content k is denoted by Q_k.
- **Caching strategy vector of EDP *i*:** $x_i(t) = [x_{i,1}(t), \cdots, x_{i,K}(t)]$ → the instantaneous caching rate of contents
- **Remaining Storage capacity of *i*:** $q_i(t) = [q_{i,1}(t), \cdots, q_{i,K}(t)]$

System Model

- **Network Model:** the evolution of $h_{i,j}(t)$ → the channel fading coefficient between EDP i and requester j

$$\mathbf{d}h_{i,j}(t) = \frac{1}{2}\varsigma_h(\upsilon_h - h_{i,j}(t))\mathbf{d}t + \varrho_h \mathbf{d}W_{i,j}(t).$$

ς_h: changing rate $W_{i,j}(t)$: a standard Brownian motion
υ_h& ϱ_h: the long-term mean and standard deviation

- **Edge Caching Model**: Content Popularity + Timeliness → The dynamics of the caching state

$$\mathbf{d}q_{i,k}(t) = Q_k[-w_1 x_{i,k}(t) - w_2 \Pi_{i,k}(t) + w_3 \xi^{L_{i,k}(t)}]\mathbf{d}t + \varrho_q \mathbf{d}W_i(t)$$

Content popularity: the frequency at which the content k is requested

Content timeliness: the level of urgency with which requesters acquire content k

The fewer the requests for content k, the faster it is removed from the caching storage.
The more urgent the requests for content k, the faster it is added into the caching storage.

- **Trading Model**: $p_{i,k}(t)$→ the unit price customized by EDP i for selling content k
  Sharing price $\bar{p}_k$→ a uniform unit price for obtaining content k from other peer EDPs

Game Formulation

- **Three cases when trading**

Case 1
EDP i has already cached enough content k → $\mathbb{P}^1$

Case 2
EDP i buys content k from some adjacent EDPs → $\mathbb{P}^2$

Case 3
EDP i needs to download uncached content k → $\mathbb{P}^3$

- **Trading income:** the dynamic price * the number of requesters

$$\Phi^1_{i,k}(t) = I_{i,k}(t) p_{i,k} \mathbb{P}^1 \left(Q_k - q_{i,k}(t)\right) + I_{i,k}(t) p_{i,k} \mathbb{P}^2 \left(Q_k - q_{-i,k}(t)\right) + I_{i,k}(t) p_{i,k} \mathbb{P}^3 Q_k$$

The set of requesters who ask for content k at time t

$$p_{i,k}(t) = \begin{cases} \hat{p}, & M = 1 \\ \hat{p} - \dfrac{\eta_1 \sum_{i'=1, i' \neq i}^{M} Q_k \cdot x_{i',k}(t)}{M-1}, & M \geq 2 \end{cases}$$

The principle of supply and demand

Sharing benefit: the monetary benefit formed when EDP i shares its contents with other EDPs $\mathcal{M}_{i,k}(t)$

$$\Phi^2_{i,k}(t) = \sum_{i' \in \mathcal{M}_{i,k}(t)} \overline{p}_k \left(q_{i',k}(t) - q_{i,k}(t)\right)$$

Game Formulation

- **Content placement cost** $C_{i,k}^{1}(t) = w_4 x_{i,k}(t) + w_5 x_{i,k}^{2}(t)$ Placing and storing cached contents

Staleness cost: a penalty function of the total request service delay

$$C_{i,k}^{2}(t) = \eta_2 \left\{ \frac{Q_k x_{i,k}(t)}{H_c} + \sum_{j \in I_{i,k}(t)} \left[\mathbb{P}^1 \frac{Q_k - q_{i,k}(t)}{H_{i,j}(t)} + \mathbb{P}^2 \frac{Q_k - q_{-i,k}(t)}{H_{i,j}(t)} + \mathbb{P}^3 \left(\frac{q_{i,k}(t)}{H_c} + \frac{Q_k}{H_{i,j}(t)} \right) \right] \right\}$$

Sharing cost $C_{i,k}^{3}(t) = \mathbb{P}^2 \bar{p}_k \left(q_{i,k}(t) - q_{-i,k}(t) \right)$ Buy content k from an adjacent EDP

Game

$$max_{x_{i,k}(t), t:0 \to T} \mathcal{U}_{i,k}\left(x_{i,k}, S_{i,k}, S_{-i,k}\right) = \Phi_{i,k}^{1}(t) + \Phi_{i,k}^{2}(t) - C_{i,k}^{1}(t) - C_{i,k}^{2}(t) - C_{i,k}^{3}(t)$$

$$\boldsymbol{s.t.}, \quad \mathbf{d}h_{i,j}(t) = \frac{1}{2} \varsigma_h (\upsilon_h - h_{i,j}(t))\mathbf{d}t + \varrho_h \mathbf{d}W_{i,j}(t)$$

$$\mathbf{d}q_{i,k}(t) = Q_k[-w_1 x_{i,k}(t) - w_2 \Pi_{i,k}(t) + w_3 \xi^{L_{i,k}(t)}]\mathbf{d}t + \varrho_q \mathbf{d}W_i(t)$$

Outline

- Background & Motivation
- Related Work & Problem Formulation
- Basic Idea & Solution
- Evaluation & Conclusion

Basic Idea

- Solving a series of Hamilton-Jacobi-Bellman (HJB) equations

$$max_{x_{i,k}(t)}\left[\mathcal{O}\mathcal{V}_{i,k}(t) + \mathbf{U}_{i,k}(t)\right] + \partial_t \mathcal{V}_{i,k}(t) = 0$$

Partial differential operator

$$\mathcal{V}_{i,k}\left(t, S_{i,k}, S_{-i,k}\right) = max_{x_{i,k}(t)} \mathcal{U}_{i,k}\left(x_{i,k}, S_{i,k}, S_{-i,k}\right)$$

There exists a complicated coupling of the M equations;
Unknown mutual influence: a dynamic pricing policy and a peer-to-peer sharing policy.

Approximate the **collective impact** of large-scale EDPs on caching, trading, and sharing, and then iteratively solve multiple coupled equations to determine **the optimal caching strategy** for each EDP.

The original stochastic differential game → A Mean-Field Game with a generic-player representation

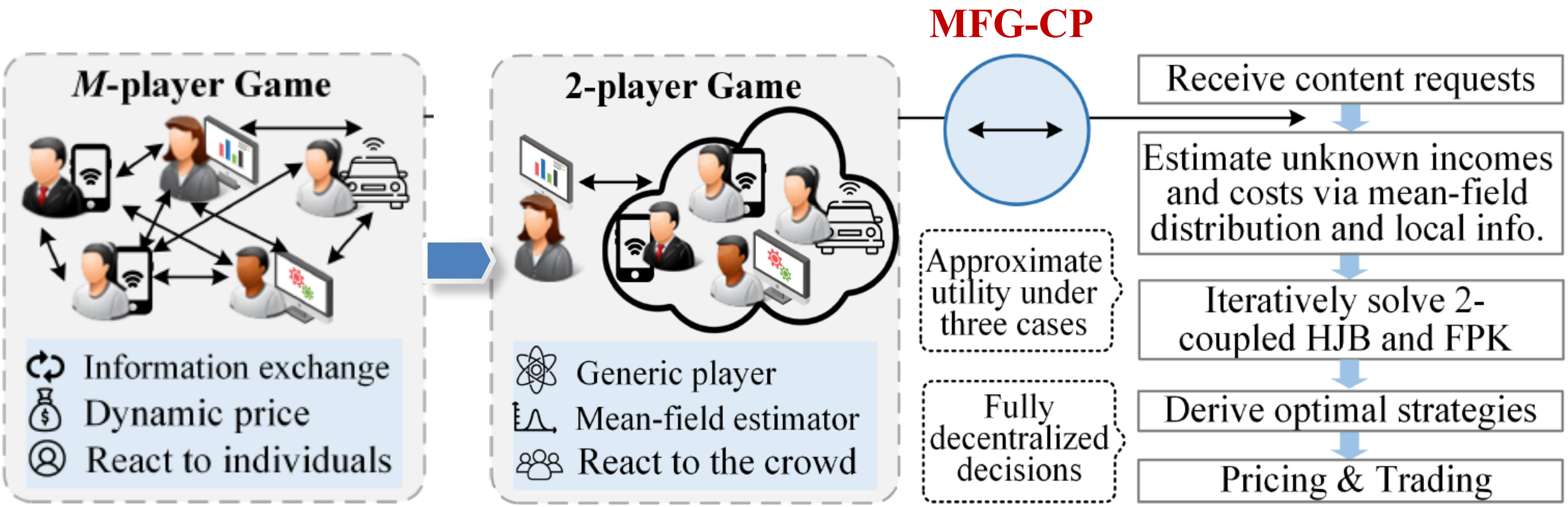
Estimate

A mean-field estimator → Estimate the unknown incomes & costs of EDPs
A generic player → Adjust its caching strategy

Solve

An iterative best response learning algorithm to solve the coupled equations in a distributed way

Solution



- ✓ The number of partial differential equations in the MFG-CP is reduced from M×K to 2×K
  → The computing time required to derive the optimal strategies will not incur excessive complexity.
- ✓ The optimal caching strategy can be determined during the time horizon using **only local information and a mean-field distribution**.

Determining the Optimal Caching Strategy

Mean-field Estimator

- Construct the Fokker-Planck-Kolmogorov (FPK) equation to estimate the statistical distribution of states $\lambda(S_k(t))$:

$$\partial_t \lambda(S_k(t)) + \frac{1}{2}\varsigma_h(\upsilon_h - h(t))\partial_h \lambda(S_k(t)) + Q_k[-w_1 x_k(t) - w_2 \Pi_k(t) + w_3 \xi^{L_k(t)}]\partial_q \lambda(S_k(t)) - \frac{1}{2}\varrho_h^2 \partial_{hh}^2 \lambda(S_k(t)) - \frac{1}{2}\varrho_q^2 \partial_{qq}^2 \lambda(S_k(t)) = 0.$$

- Estimate the dynamic price:

$$p_k(t) \approx \hat{p} - \eta_1 Q_k \int_h \int_q \lambda(S_k(t)) x_k^*(S_k(t)) \mathbf{d}h\mathbf{d}q$$

- Estimate the average caching state:

$$\bar{q}_{-,k}(t) \approx \int_h \int_q q_k(t)\lambda(S_k(t))\mathbf{d}h\mathbf{d}q_k$$

- Estimate the average sharing benefit:

$$\overline{\Phi}_k^2(t) = \overline{p}_k \overline{\Delta q}(t)\left(\frac{M - M_k'(t)}{M_k(t)} - 1\right)$$

λ

x

Generic player

- Solve the optimization problem:

$$\max_{x_k(0\to T)} \quad \mathcal{U}_k(x_k, S_k, \lambda)$$

- Employ the mean-field estimator to quickly obtain the trading income, sharing benefit, staleness cost, and sharing cost.
- Reconstruct HJB equation to determine the value function $\mathcal{V}_{i,k}(t)$:

$$\max_{x_k(t)} \Big[\frac{1}{2}\varsigma_h(\upsilon_h - h(t))\partial_h \mathcal{V}_k(t) + \frac{1}{2}\varrho_h^2 \partial_{hh}^2 \mathcal{V}_k(t) + Q_k[-w_1 x_k(t) - w_2 \Pi_k(t) + w_3 \xi^{L_k(t)}]\partial_q \mathcal{V}_k(t) + \frac{1}{2}\varrho_q^2 \partial_{qq}^2 \mathcal{V}_k(t) + \mathbf{U}_k(t, x_k, S_k, \lambda)\Big] + \partial_t \mathcal{V}_k(t) = 0.$$

- The optimal caching strategy:

$$x_k^*(t) = \left[-\left(\frac{w_4}{2w_5} + \frac{\eta Q_k}{2H_c w_5} + \frac{Q_k w_1 \partial_q \mathcal{V}_k(t)}{2w_5}\right)\right]^+$$

Iterative Learning

Algorithm 2: Iterative Best Response Learning Scheme

input : The iterative number ψ; iterative threshold ψ_{th};
output: The optimal caching strategy;

Initialize: $\psi = 0$; $x_k^{\psi}(0)$; $\mathcal{V}_k^{\psi}((\sigma+1)T)$; λ;
while $\psi < \psi_{th}$ **do**
    $\psi = \psi + 1$;
    Solve the HJB equation using $x_k^{\psi-1}(t)$ and λ;
    Update the strategy vector $x_k^{\psi}(t)$ based on Eq. (21);
    if $|x_k^{\psi}(t) - x_k^{\psi-1}(t)| <$ *a preset threshold* **then**
        Break the iterative learning process;
    Solve the FPK equation using $x_k^{\psi}(t)$ to update λ;
    Update the mean-field estimator: $p_k(t), \bar{q}_{-,k}(t), \overline{\Delta q}(t)$;
    Update the utility function according to Eq. (10);
return $x_k^{\psi}(t)$ for $t \in [\sigma T, (\sigma+1)T)$.

Core: **iteratively solve the coupled HJB equation and FPK equation**: the backward HJB models the induction process of the optimization of each individual, while the forward FPK models the evolution of the mean-field as a whole.

The solution of the HJB equation has a great impact on the FPK equation to update the mean field.

The solution of the FPK equation is also required by the HJB equation to estimate the generic player's utility and update strategies.

Framework Description

Algorithm 1: The Proposed Framework (MFG-CP)

input : The parameters $w_1 \sim w_5, \eta_1, \eta_2, \varsigma_h, \upsilon_h, \varrho_h, \varrho_q$, and ξ; the maximum epoch number σ_{max};

Initialize: the state $S(0)$; the data size of each content Q_k;

for *each EDP* $i = 1, 2, \cdots, M$ *in parallel* **do**

  while *each optimization epoch* $\sigma \leq \sigma_{max}$ **do**

    Record requests $\{I_{i,k}(t) | k \in \mathcal{K}, t \in [\sigma T, (\sigma+1)T)\}$;

    Determine the content set $\mathcal{K}' = \{k | q_{i,k}(t) < Q_k$ and $\sum_{\sigma T}^{(\sigma+1)T} |I_{i,k}(t)| > 0\}$ that needs to be cached;

    for *each content* $k \in \mathcal{K}'$ **do**

      // **Mobile Edge Caching**;

      Compute content popularity $\Pi_{i,k}(t)$ and content timeliness $L_{i,k}(t)$ for all contents;

      Call for **Alg. 2** to obtain the optimal caching strategy $x^*_{i,k}(t)$ as its best response;

      Update the current state based on $x^*_{i,k}(t)$;

      // **Pricing and Trading**;

      Case 1: sell the content at a unit price $p_{i,k}(t)$;

      Case 2: buy the uncached content from an adjacent EDP at $\bar{p}_k$ and sell it to requesters;

      Case 3: download the uncached content from the center and sell it to requesters;

Core: Each EDP makes decisions in parallel

→ **Mobile Edge Caching + Pricing and Trading**

The change in requesters' demands occurs at a relatively slow rate

Invokes Alg. 2 to give the best response: customizing its optimal caching strategy

The trading process between the EDP and the corresponding requesters will be carried out, where the EDP needs to take different actions under various cases

Equilibrium Analysis

Lemma

- ✓ In the MFG-CP framework, there exists the **unique value function** $\mathcal{V}_k(t)$ of the HJB equation.
  Proof: (i) The caching strategy space is a compact subset of R;
  (ii) The drift term of the state dynamics & the utility function: bounded+Lipschitz continuous.
- ✓ There exists the **unique mean-field distribution** $\lambda\big(S_k(t)\big)$ of the FPK equation.
  Proof: A parabolic partial differential equation → $\partial_t \lambda\big(S_k(t)\big) + \Theta = d$

  $\Theta = -\sum_{i,j=1}^{k} a_{i,j}\lambda\big(S_i S_j\big) + \sum_{i,j=1}^{k} b_i \lambda(S_i) + c\lambda$ → $a_{i,j} = a_{j,i}$

Theorem

- ✓ In the MFG-CP framework, there exists a **unique Nash equilibrium**.
  Proof:

  Given this contraction mapping, there exists a unique fixed-point based on the fixed-point theorem → The unique solution pair of the HJB-FPK equations

Outline

- Background & Motivation
- Related Work & Problem Formulation
- Basic Idea & Solution
- Evaluation & Conclusion

Evaluation

Experimental Settings

Real Dataset

- YouTube → The number of requests for each category → content id, tags, views, comment count, description.

Parameter settings

- M=300 EDPs, K=20 contents
- The conversion parameter η change from [0.1, 0.4]
- $\lambda(0) \sim \mathcal{N}(0.7, 0.1^2)$

Compared Algorithms

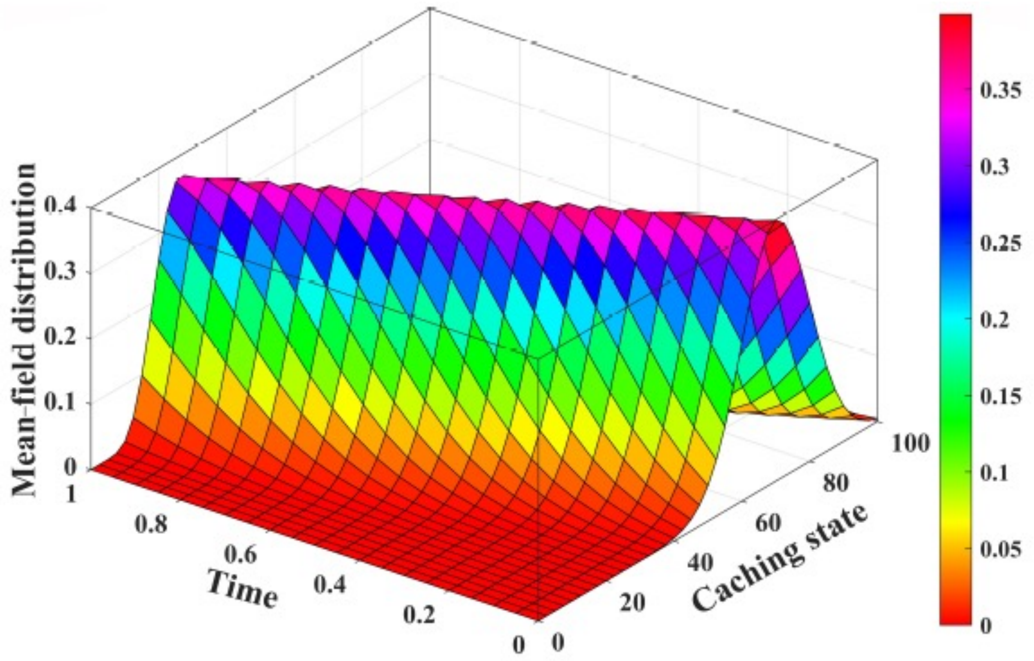
- Random Replacement (RR)
- Most Popular Caching (MPC)
- MFG: do not consider sharing
- Ultra-Dense Caching Strategy (UDCS): consider content overlap

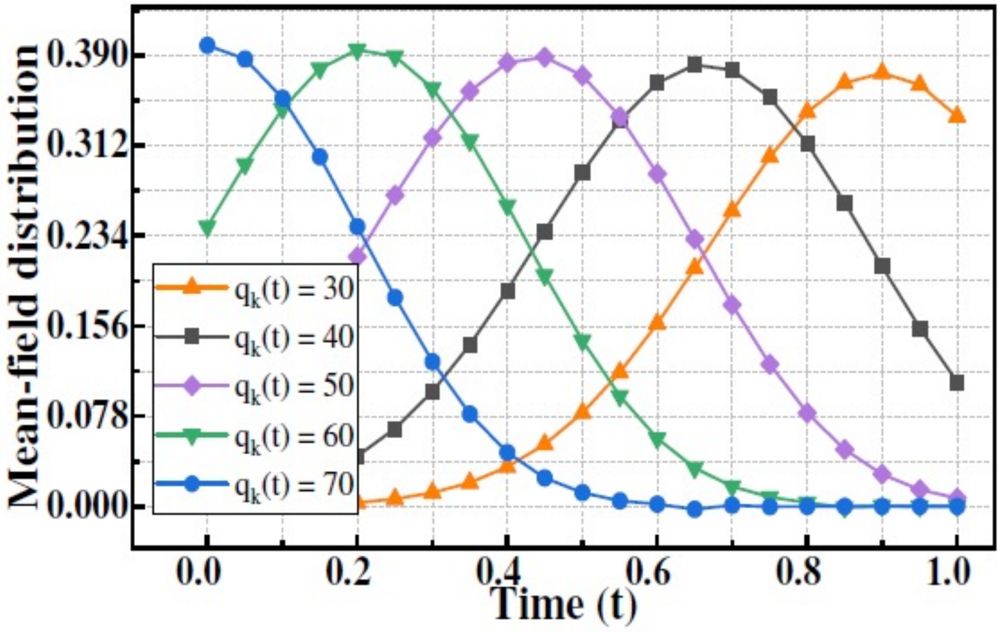
Evaluation Metrics

- Mean-Field Equilibrium
- Caching state & Utility
- Computation time

Evaluation

- Evolution of the mean-field distribution at the equilibrium:



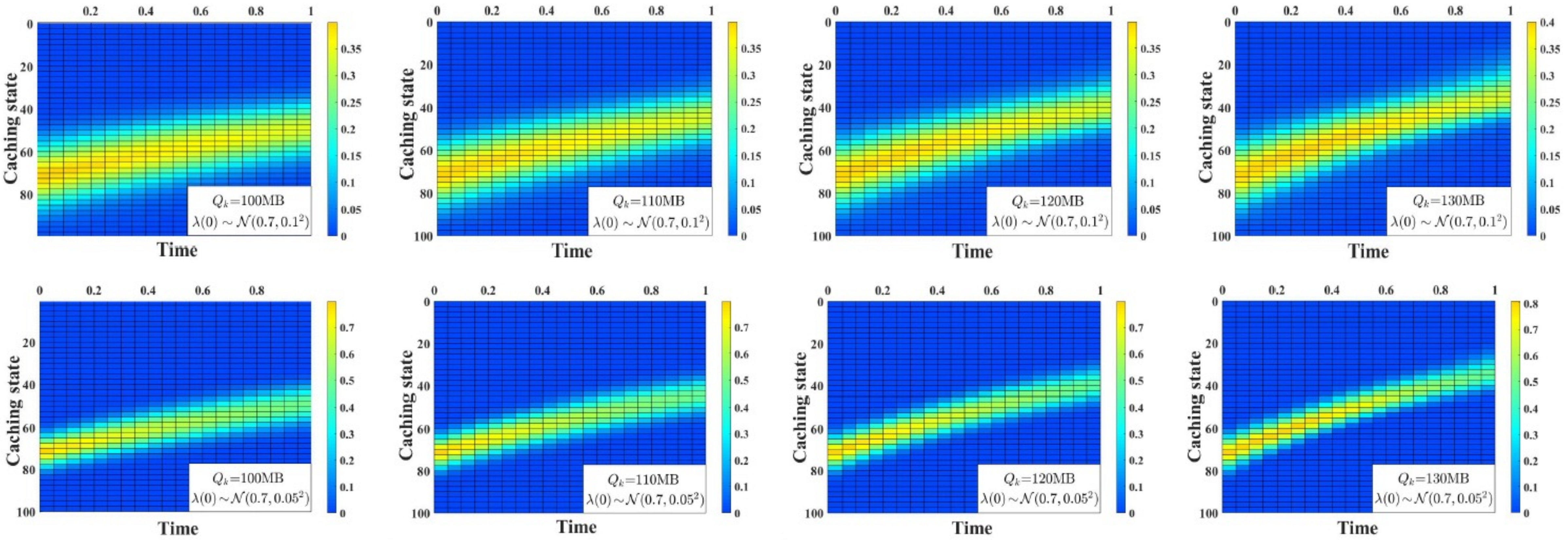


When we fix the time slot, the size of the remaining caching space will increase first and then decrease.

As the time evolves, the remaining caching space with {60MB, 70MB} will vanish due to the improvement of space utilization.

Evaluation

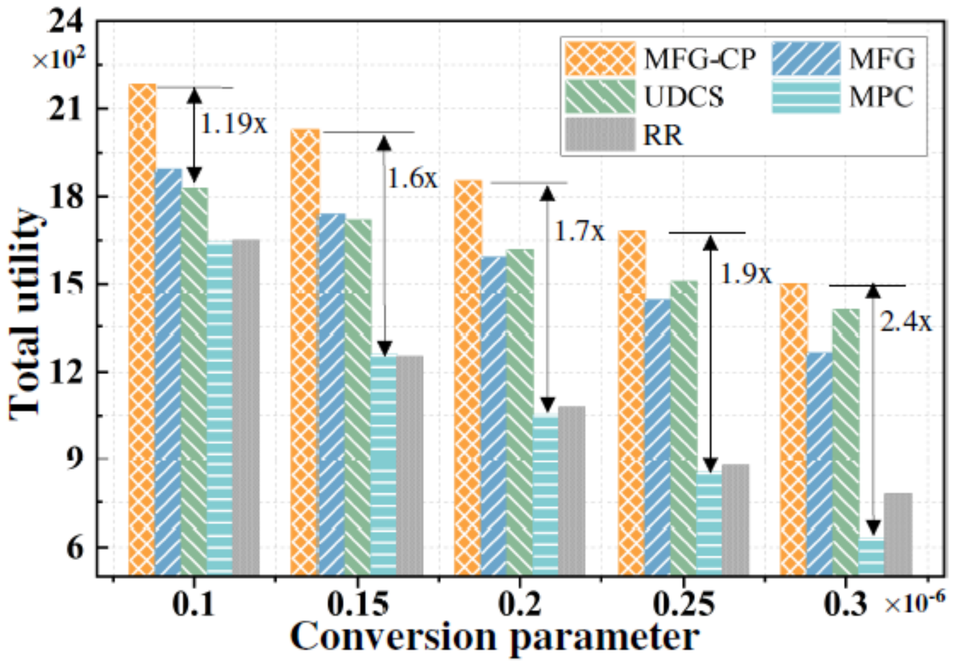
- A heat map description of the mean-field distribution under different Q_k:

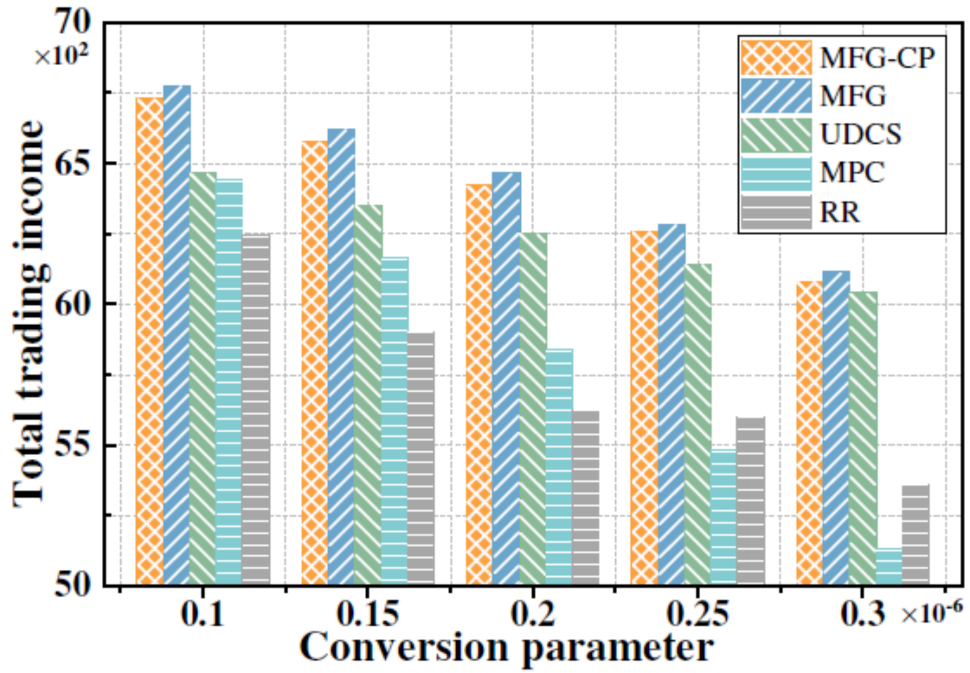


The caching space will gradually reach saturation with the increase of Q_k.
When decreasing the variance values, the heat map displays more concentrated results.

Evaluation

- Comparisons under different conversion parameters:

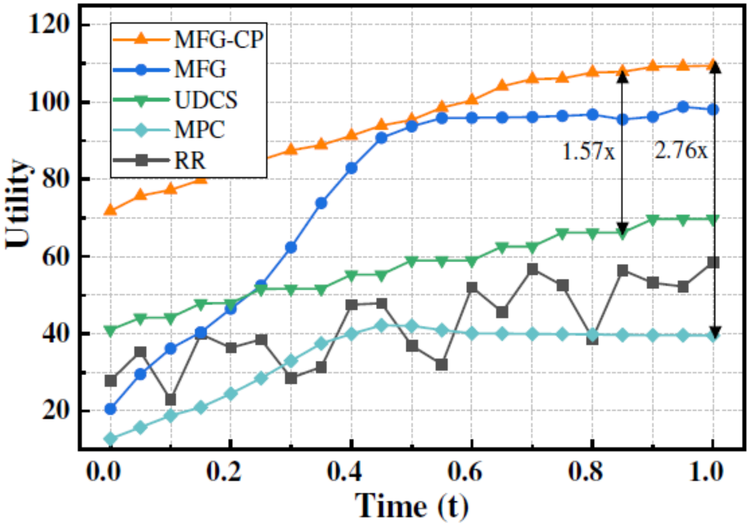


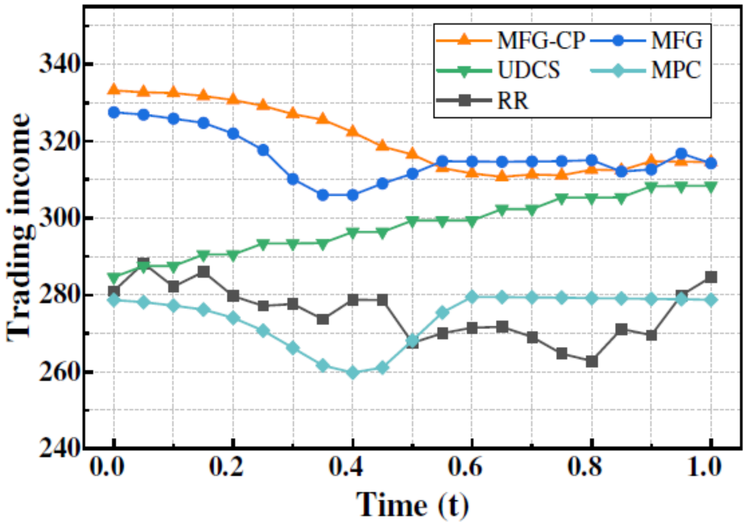


- Improving the value of the conversion parameter leads to a reduction in total utility;
- The staleness cost of MFG obviously exceeds that of MFG-CP;
- The behavior of content sharing is beneficial for improving the utility of each EDP;
- MFG-CP enables each EDP to possess a higher utility than these compared algorithms.

Evaluation

- Comparisons on utility, trading income, and time:





The proposed framework MFG-CP offers significant advantages in maximizing the utilities of EDPs

COMPARISONS ON COMPUTATION TIME (SECOND)

Methods \ Number	50	100	200	300
MFG-CP	0.4319	0.4442	0.4336	0.5121
RR	0.1697	0.5527	0.9766	1.7832
MPC	0.1657	0.3157	0.8694	1.7094

The computational complexity of MFG-CP does not increase with the number of EDPs

Conclusion

01

✓ We study the competitive content placement issue in large-scale dynamic MEC systems.

02

✓ We model the problem as a non-cooperative stochastic differential game, which considers the heterogeneous content demands, real-time trading prices, and paid content sharing.

03

✓ To facilitate decentralized decision-making, we propose the MFG-CP framework for joint content caching and pricing.

04

✓ We develop an iterative best response learning scheme to determine the optimal caching strategy for each EDP.

05

✓ Extensive simulations on real-world traces validate its great performance.

中国科学技术大学
University of Science and Technology of China

40th IEEE International Conference on
Data Engineering

Thank you for your attention!

Question?

Yin Xu xuyin218@mail.ustc.edu.cn