DOMINATION STRATEGIES FOR FREE-RIDING IN CROSS-SILO FL-BASED CACHING

Jiqing Gu[1,2], *Chao Song*[1,*], *Jianfeng Huang*[1], *Jie Wu*[3], *Ruilin Hu*[1] *and Li Lu*[1]

[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China
[2]School of Computer Science, Chengdu University of Information Technology
[3]Department of Computer and Information Sciences, Temple University

ABSTRACT

Federated Learning (FL) has been widely applied to content popularity prediction in caching systems to enhance Quality of Experience (QoE). In cross-silo FL, distributed organizations with local caches collaboratively train a global model coordinated by a central server. As the global model is a public good, caches may benefit without contributing, leading to free-riding. Since local request data is time-varying, repeated FL tasks form a long-term game in which free-riding destabilizes social welfare, defined as the sum of expected payoffs of all caches. This paper proposes a multi-player multi-action zero-determinant (MMZD) strategy to dominate free-riders and control social welfare. Using the FedML framework and YouTube datasets, experiments show that MMZD achieves stable and high social welfare.

Index Terms— Content popularity prediction, cross-silo federated learning, domination strategies, social welfare control

1. INTRODUCTION

The explosive growth of content, especially videos, has driven the development of caching technologies supported by Content Providers (CPs, e.g., YouTube, TikTok) and Internet Service Providers (ISPs). Caching reduces transmission delays and improves Quality of Experience (QoE) by proactively storing popular content, which typically follows the Zipf-law distribution [1–3]. By caching highly popular content in local servers, ISPs can offload network traffic and allow users to fetch content without contacting the original server.

However, due to business competition and privacy concerns, ISPs are unwilling to share raw data for centralized training. Federated Learning (FL) provides a privacy-preserving alternative that enables collaborative model training across ISPs without exposing local data [4–6]. In cross-silo FL, each ISP trains locally while a central server coordinates aggregation, and the resulting global model is jointly owned. Since the model is a public good—non-excludable and non-rivalrous—caches may benefit without equivalent contribution, giving rise to the free-riding problem [7]. As illustrated in Fig. 1, a free-rider can skip uploading its local model but still obtain the global one, reducing accuracy and lowering the collective payoff. This behavior degrades prediction performance, reduces social welfare, and destabilizes repeated FL tasks.

We define social welfare as the sum of expected payoffs of all caches, and cooperation among caches optimizes the social welfare in cross-silo FL. However, during this cooperation, some caches may cheat to gain a higher payoff, resulting in unsatisfactory social welfare for the others. This leads to selfish behavior where caches, instead of fully participating in local training, contribute minimally

* Corresponding author

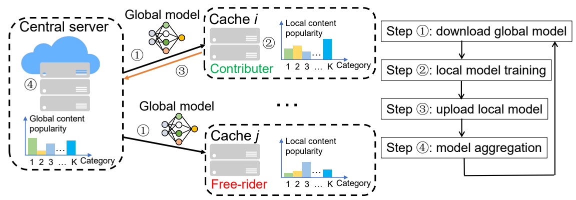


Fig. 1. Content popularity prediction in cross-silo FL-based caching.

but still benefit from the global model. This phenomenon is called free-riding [7], e.g., cache j in Fig. 1 is a free-rider that does not upload its local model. Free-riding results in a non-optimal global model, which lowers social welfare and harms the long-term stability and sustainability of cross-silo FL tasks. Such cross-silo FL is a repeated game, with caches cooperating daily to train the global model for content popularity prediction. However, due to free-riding, social welfare fluctuates, becoming unstable and suboptimal.

To address free-riding and maintain stable social welfare at a maximal possible value, we should resolve two challenges: (1) As the number of caches increases, the action space becomes large, leading to unstable social welfare. How can we achieve stable and maximal social welfare? (2) By considering the existence of free-riding, how to incentivize the caches under the cross-silo federated learning for content popularity prediction?

In this paper, we propose a multi-player multi-action zero-determinant strategy (MMZD) to control social welfare and mitigate free-riding. The zero-determinant strategy is a class of probabilistic and conditional strategies [8, 9], where a player unilaterally sets the sum of expected payoffs (social welfare) or adjusts the payoff ratio between themselves and their opponents, independent of their opponents' strategies. We model content popularity prediction in cross-silo FL as a repeated game characterized by a Markov chain. In each iteration, one cache adopts the MMZD strategy, while others decide how many rounds of global aggregation they will participate in. The cache adopting MMZD can control the social welfare unilaterally, maintaining it at a stable and maximal value. This is done by performing extra aggregations based on the involvement of participants, with free-riders contributing less. We also define the controllable range of social welfare and discuss its influence with MMZD. Our main contributions are summarized as follows:

(1) We utilize MMZD to control social welfare and mitigate free-riding in cross-silo FL, maintaining a stable and maximal value of social welfare.

(2) We provide a theoretical analysis of social welfare control in repeated cross-silo FL, deriving the controllable range and explaining its role in ensuring stability against free-riding.

(3) We validate the proposed MMZD strategy through numerical analysis and experiments, demonstrating its effectiveness in controlling social welfare within the derived range.

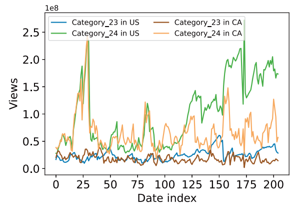


Fig. 2. Time-varying requests.

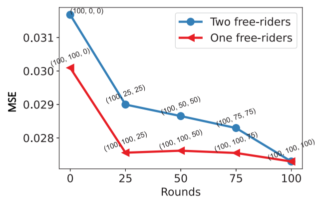
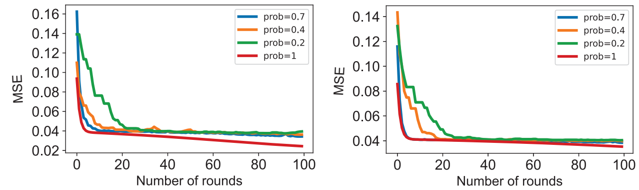


Fig. 3. Influence of free-riding.



(a) Prediction of views.

(b) Prediction of comment counts.

Fig. 4. Performance of global model when free-riders exist.

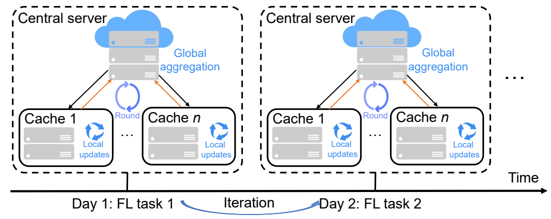


Fig. 5. Repeated game of cross-silo FL among multiple local caches.

2. SYSTEM MODEL

2.1. Caching and Cross-silo FL Model

Caching improves user experience by storing popular content on cache servers, reducing congestion and delays. Different ISPs own caches with heterogeneous request patterns, as shown in Fig.2, where YouTube video demands vary across regions[1]. Since local records are limited, accurate prediction requires collaboration across caches. To preserve privacy, we adopt a cross-silo Federated Learning (FL) framework [10], in which a central server coordinates training while caches perform local updates and share parameters for aggregation, as illustrated in Fig.1. Each cache executes K local updates and may join r global aggregations selectively due to self-interest, with its contribution represented by an action vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)$, where x_i denotes the number of global aggregations that cache i participates in during the FL task.

2.2. Motivation

In cross-silo Federated Learning (FL), caches can benefit from the global model without contributing, leading to free-riding. This happens when caches participate minimally in local training but still gain from the trained model. We conducted an experiment with three caches in cross-silo FL, as shown in Fig. 3, where the number near each marker indicates the rounds of participation in global aggregation. The results demonstrate that prediction error (MSE) increases when more caches behave as free-riders, confirming their negative impact on model performance. We further evaluated different participation probabilities (e.g., 0.2, 0.4, 0.7, 1) in each aggregation round. As shown in Figs. 4(a) and 4(b), prediction accuracy consistently improves with higher participation probabilities, achieving the best performance when all caches fully cooperate (prob = 1). These findings indicate that free-riding leads to a non-optimal global model, lowers social welfare, and threatens the long-term stability of cross-silo FL tasks. Therefore, controlling free-riding behavior is essential to ensure stable social welfare in FL-based caching systems.

2.3. Payoff of the Cache

In cross-silo FL, all caches receive the same global model after each FL task. The payoff of cache i is defined as: $U_i(\boldsymbol{x}) = f_i(\boldsymbol{x}) - g_i(\boldsymbol{x})$, where $f_i(\boldsymbol{x})$ and $g_i(\boldsymbol{x})$ represent the income and cost of cache i, respectively. Inspired by [11], we define the income function as: $f_i(\boldsymbol{x}) = o_i(\rho(\mathbf{0}) - \rho(\boldsymbol{x}))$, where o_i is the unit revenue of cache i, $\rho(\mathbf{0})$ is the loss of the untrained global model, $\rho(\boldsymbol{x})$ is the loss of the trained model with action vector $\boldsymbol{x}$ and modeled as $\rho(\boldsymbol{x}) = \frac{\theta_0}{\theta_1 + K\sum_{i\in\mathcal{N}} x_i}$, where θ_0 and θ_1 are coefficients determined by the loss function, neural network, and local datasets. The income of cache i is proportional to the difference in loss between the untrained and trained global models. Increasing participation rounds reduce the theoretical loss $\rho(\boldsymbol{x})$, with diminishing marginal decreases. The cost of cache i is defined as: $g_i(\boldsymbol{x}) = \mu_i K x_i + C^m$, where μ_i is the computation cost per iteration in cache i's local training, K is the number of local updates per round, and C^m is the communication cost for uploading and downloading model updates. If cache i does not participate in global aggregation ($x_i = 0$), it only submits a zero vector. The payoff function of cache i is then:

$$U_i(\boldsymbol{x}) = o_i(\rho(\mathbf{0}) - \frac{\theta_0}{\theta_1 + K\sum_{i\in\mathcal{N}} x_i}) - (\mu_i K x_i + C^m), \quad (1)$$

where $U_i(\boldsymbol{x})$ is the payoff of cache i with action vector $\boldsymbol{x}$, used to calculate the MMZD strategy in subsection 3.1. The assumption is that the payoff of any local cache $i \in \mathcal{N}$ is negative if it only uses its local data for training a local model. Thus, this assumption motivates the caches to participate in global model training the cross-silo federated learning.

2.4. Repeated Game

The cross-silo FL game is a repeated game because caches cooperate in multiple FL tasks over time, as shown in Fig. 5. Each game iteration corresponds to a specific FL task. In this game, caches act as players, with cache i's action denoted by $x_i \in \{0, 1, 2, ..., r\}$ and its payoff as $U_i(\boldsymbol{x})$. This is a multi-player, multi-action repeated game, where we focus on long-term social welfare.

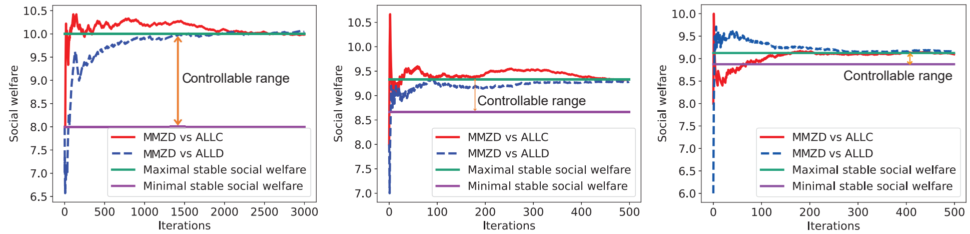
Definition 1 (**Social Welfare**). *Social welfare E_{all} is defined as the sum of expected payoffs of all caches in the long term, $E_{all} = \sum_{i\in\mathcal{N}} E_i$, where E_i is the expected payoff of cache i.*

In the repeated cross-silo FL game, free-riders cause fluctuating social welfare. Thus, we aim to avoid free-riding and stabilize social welfare at a maximal possible value, modeling this as follows:

Problem 1 (**Social Welfare Control**). *The social welfare is required to be controlled at a stable and maximal possible value, which indicates the social welfare shows little changes and maintains around a maximal possible value with increase of the game iteration.*

$$\begin{aligned} max \quad & E_{all} = \sum_{i=1}^{n} \alpha_i E_i \\ s.t. \quad & E_{all} = \gamma, \gamma \in [\gamma_{min}, \gamma_{max}] \end{aligned} \quad (2)$$

[1] https://www.kaggle.com/datasets/datasnaek/youtube-new



(a) $\lambda = 0.005$. (b) $\lambda = 0.13$. (c) $\lambda = 0.15$.

Fig. 6. Numerical results of different λ on controllable range by a study case.

where α_i refers to a weight of E_i, and γ is a variable in a range of social welfare [γ_{min}, γ_{max}].

The constraint in Eq. (2) ensures that social welfare remains constant. The objective and constraint aim to maintain social welfare at a stable and maximal value.

3. DOMINATION STRATEGY FOR FREE-RIDING

In this section, we use the Multi-player Multi-action Zero-Determinant (MMZD) strategy [12, 13] for social welfare control in cross-silo FL of caching and provide a theoretical analysis. The motivation for using MMZD stems from its ability to enable unilateral control of long-term outcomes in repeated games. In contrast to Nash-based or evolutionary strategies that require mutual cooperation or adaptive learning, MMZD allows one player to fix the expected payoff (social welfare) regardless of opponents' strategies. This is particularly valuable in federated caching systems where full cooperation cannot be guaranteed and free-riding behavior is unpredictable. MMZD empowers a cache with market power to stabilize system performance under worst-case assumptions.

3.1. MMZD Strategy

In the repeated game [14], the strategy profile determines a stochastic process, which can be modeled as a Markov chain. We define the action, state, and strategy as follows: Each cache i's action is $x_i \in \{0, 1, 2, ..., r\}$, and there are $(r+1)^n$ possible action-tuples per game iteration. The mixed strategy $\mathbf{p}^i$ for cache i is the probability distribution over actions. The utility vector $\mathbf{u}^i$ represents the utility for each possible action and outcome.

The multi-player game can be characterized by a Markov chain with a transition matrix $\mathbf{M}$. If $\mathbf{M}$ is regular, a stationary vector $\boldsymbol{v}$ exists, and the expected utility of cache i in the stationary state is: $E_i = \dfrac{\boldsymbol{v}^T \cdot \mathbf{u}^i}{\boldsymbol{v}^T \cdot \mathbf{1}}$. The total expected utility is:

$$\sum_{i=1}^{n} \alpha_i E_i - \gamma = \frac{\det(\mathbf{p}^1, \mathbf{p}^2, ..., \mathbf{p}^n, \sum_{i=1}^{n} \alpha_i \mathbf{u}^i - \gamma \mathbf{1})}{\det(\mathbf{p}^1, \mathbf{p}^2, ..., \mathbf{p}^n, \mathbf{1})}. \quad (3)$$

If cache i adopts a zero-determinant strategy, it can control the social welfare to a fixed value γ, independent of other caches' strategies. For example, with $\alpha_i = 1$, the social welfare is $\sum_{i=1}^{n} \alpha_i E_i = \gamma$, and the optimal social welfare is obtained by solving below constrained optimization problem:

$$max \quad \gamma = \sum_{i=1}^{n} \alpha_i E_i,$$
$$s.t. \begin{cases} 0 \le p^1_{j,s} \le 1, j \in \{0, 1, 2, ..., r\}, s \in \{1, 2, ..., (r+1)^n\} \\ \hat{\mathbf{p}}^1 = \lambda(\sum_{i=1}^{n} \alpha_i \mathbf{u}^i - \gamma \mathbf{1}), \\ \lambda \neq 0. \end{cases} \quad (4)$$

Here, λ controls the convergence rate to the stable state. When $\lambda > 0$, we can obtain

$$\gamma_{max} = \min\{\frac{1}{\lambda} + \sum_{i=1}^{n} u^i_1, ..., \frac{1}{\lambda} + \sum_{i=1}^{n} u^i_{(r+1)^{n-1}}, \sum_{i=1}^{n} u^i_{(r+1)^{n-1}+1}, ..., \sum_{i=1}^{n} u^i_{(r+1)^{n+1}}\}. \quad (5)$$

When $\lambda < 0$, we can obtain

$$\gamma_{max} = \min\{\sum_{i=1}^{n} u^i_1, ..., \sum_{i=1}^{n} u^i_{(r+1)^{n-1}}, -\frac{1}{\lambda} + \sum_{i=1}^{n} u^i_{(r+1)^{n-1}+1}, ..., -\frac{1}{\lambda} + \sum_{i=1}^{n} u^i_{(r+1)^{n+1}}\}. \quad (6)$$

Thus, the cache 1 adopts the MMZD strategy $\mathbf{p}^1$ and can control the expected social welfare at a stable and maximal value γ_{max}, where $\mathbf{p}^1$ satisfies $\hat{\mathbf{p}}^1 = \lambda(\sum_{i=1}^{n} \alpha_i \mathbf{u}^i - \gamma \mathbf{1})$, and each element in $\mathbf{p}^1$ is calculated by:

$$p^1_m = \begin{cases} \lambda(\sum_{i=1}^{n} u^i_m - \gamma_{max}) + 1, m = 1, 2, ..., (r+1)^{n-1}, \\ \lambda(\sum_{i=1}^{n} u^i_m - \gamma_{max}), m = (r+1)^{n-1}+1, ..., (r+1)^{n+1}, \end{cases} \quad (7)$$

where p^1_m refers to the m-th element in $\mathbf{p}^1$.

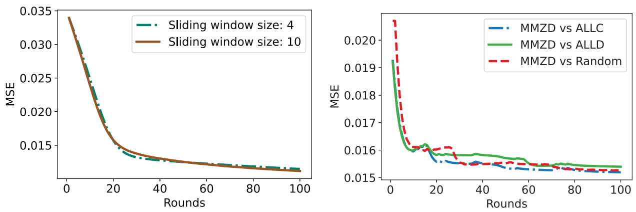
In practice, if cache 1 is chosen to adopt the MMZD strategy, cache 1 has the following prior knowledge: (1) The number of global aggregations r of an FL task before the repeated game starts and (2) the payoff functions of each cache. Cache 1 utilizes its prior knowledge to calculate each element (probability) of the MMZD strategy, then it selects the action based on the MMZD strategy in the repeated FL game. That is, it selects how many global aggregations it will take to participate in an FL task by a probability.

3.2. Controllable Range of MMZD

Actually, the MMZD strategy that cache i adopts can control social welfare at a value γ, which is a range. We define the controllable range of social welfare by MMZD as follows:

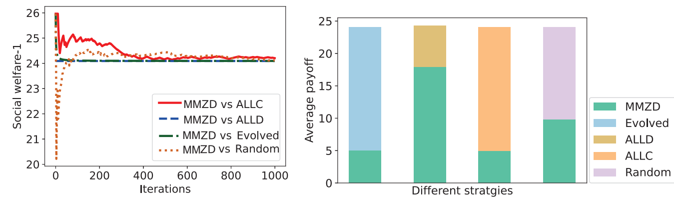
Definition 2 (**Controllable Range of Social Welfare**). *The social welfare is controlled at γ by the MMZD adopter, and γ has a maximal value γ_{max} and a minimal value γ_{min}. We denote the controllable range of social welfare as $[\gamma_{min}, \gamma_{max}]$.*

Due to the range of social welfare control (e.g., $[\gamma_{min}, \gamma_{max}]$), it is not easy to be expressed by the equation. We present the numerical analysis of the controllable range for a better understanding. For simplicity, we use a two-player two-action game as a study case. We define two players X and Y with two actions $\{1, 50\}$, which refer to a low and a high number of participation in the global aggregation. The payoff matrix is $[1, 6, 3, 5.5]$, $[1, 2, 7, 5]$ under the action-tuples



(a) Various sizes of sliding window. (b) Different strategies.

Fig. 7. Global model performance of popularity prediction.



(a) Social welfare control under MMZD strategy. (b) Cooperation probability of the evolved cache under pinning strategy.

Fig. 8. Average payoff with different strategies.

$\{(1,1),(1,50),(50,1)(50,50)\}$. The zero determinant strategy is calculated by Eq. (7). Some values of $\lambda > 0$ are feasible in this case. We choose λ as 0.005, 0.13, and 0.15 in the numerical analysis. As shown in Fig. 6, when λ is set as different values, the controllable range is different. The controllable range of Fig. 6(a) is $[8, 10]$. While in Fig. 6(b) and Fig. 6(c), the controllable ranges are in $[8.65, 9.35]$ and $[8.8, 9.25]$, respectively. This indicates that the controllable range narrows when λ increases to a certain value. Furthermore, we can see that λ with a larger value will speed up the convergence rate. When $\lambda = 0.005$, the social welfare becomes stable at about 2000 iteration in Fig. 6(a), but in Fig. 6(b), the social welfare becomes stable at about 400 iterations; in Fig. 6(c), the social welfare becomes stable at about 300 iterations.

4. EXPERIMENT

In our experiment, we utilize FedML[2], which is an open research library for federated learning. We set $K = 200$, $r = 100$, $\xi = 0.2$, and $\eta = 0.0003$. Inspired by [11], $\theta_0 = 23272$ and $\theta_1 = 50194$ are obtained from the dataset. λ is set differently in social welfare control. We apply Long Short Term Memory (LSTM) for the local model training and aggregate the global model by the FedAvg algorithm [15] to verify MMZD strategy.

We use the public dataset of YouTube[3] in the experiment. It includes the records of each day's popular videos for about one year of 10 countries, so we set $n = 10$, which is reasonable because the number of caches in cross-silo FL is small in general [16]. Each cache contains a time series of local user request records for daily popular YouTube videos, categorized by region and content type. The training data includes features such as video category, daily view count, comment count, and publication timestamp. Using a sliding time window, all cache servers collaboratively train a global model for predicting popular content the next day without sharing raw data.

Comparison Methods. (1) **ALLC** [17, 18]: All cooperation strategy means the caches always choose action r rounds for global aggregation. (2) **ALLD** [17, 18]: It is an all-defection strategy. That is, the cache does not perform local training; it submits a zero vector for global model aggregation. (3) **Random** [17, 18]: A random strategy is that a cache randomly selects a number by the probability $\frac{1}{r+1}$ from $\{0, 1, 2, ..., r\}$, which is treated as the rounds of participating in the global model aggregation. (4) **Evolved** [8]: An evolutionary player starts to choose a random action, and randomly selects an action form $\{0, 1, ..., \frac{\lfloor r \rfloor}{2}\}$. If the average payoff in the current round is larger than that of the previous round, then the probability of this action under this state increases. Otherwise, it will decrease. In the evolved strategy, we define the action in $\{0, 1, ..., \frac{\lfloor r \rfloor}{2}\}$ and $\{\frac{\lfloor r \rfloor}{2} + 1, ..., r\}$ as defection and cooperation, respectively. It will be stable with the iteration increases.

[2] https://github.com/FedML-AI/FedML

[3] https://www.kaggle.com/datasets/datasnaek/youtube-new

Performance of content prediction on FL. Fig. 7 shows the performance of the LSTM model applied to the prediction of content prediction in FL; the size of the sliding window refers to the number of past days in the sequential data. A lower MSE value reflects better performance. Fig. 7(a) shows the MSE on the test data with the size of the sliding window set as 4 and 10, respectively. On the whole, MSE decreases as the number of rounds increases. The final value of loss of the sliding window set at 10 is a little smaller than that at 4, which indicates the LSTM is valid, and longer past sequential request data contributes greater to the global model. Fig. 7(b) shows the performance of the prediction model when one cache adopts the MMZD strategy and other caches adopt different strategies. MSE decreases as the rounds increase, which indicates that the MMZD can coordinate caches with other strategies for model training and obtain a good performance.

Experimental results on social welfare control. We choose one cache to adopt MMZD, and other caches adopt one kind of strategy chosen from ALLC, ALLD, Evolved, and random strategies. As shown in Fig. 8(a), the social welfare tends to be stable around 24, which is the stable and maximal value that MMZD can control. The insight behind MMZD is that the cache adopted MMZD has a higher probability of participating in the global aggregation for maximum rounds if most of the rest of the caches are free-riders. Thus, MMZD tries its best to maintain the social welfare at a possible maximal value in the worst case. Fig. 8(b) shows the average payoffs of different strategies. Clearly, it infers that the contribution of the MMZD adopter is different when other caches adopt different strategies. When caches adopt ALLD, the MMZD adopter contributes much to sustain their total payoffs. The average payoffs of caches adopted evolved, and ALLC strategies are similar, which infers the evolved strategy tends to the ALLC in the end.

5. CONCLUSION

In this paper, we investigate the repeated game of cross-silo FL for content popularity prediction. By taking the advantage of market power, we utilize the MMZD strategy to dominate the free-riders and control social welfare. We discuss the controllable range of MMZD by numerical analysis and verify that the content prediction model is feasible in FL when caches adopt different strategies. Experiment results show that the MMZD strategy can control the social welfare to achieve a high and stable value.

Acknowledgments

The work of UESTC team was supported by the National Natural Science Foundation of China under Grant No. 62472070, 82241060, 62020106013; Sichuan Science and Technology Program under Grant No. 2024NSFSC0492; Sichuan Provincial Higher Education Talent Development Quality and Teaching Reform Project under Grant No. JG2024-0001 and JG2024-0939.

6. REFERENCES

[1] Stefano Traverso, Mohamed Ahmed, Michele Garetto, Paolo Giaccone, Emilio Leonardi, and Saverio Niccolini, "Temporal locality in today's content caching: Why it matters and how to model it," *ACM SIGCOMM Computer Communication Review*, vol. 43, no. 5, pp. 5–12, 2013.

[2] Sajad Mehrizi, Anestis Tsakmalis, Shahram ShahbazPanahi, Symeon Chatzinotas, and Björn Ottersten, "Popularity tracking for proactive content caching with dynamic factor analysis," in *Proc. of IEEE ICCC*, 2019.

[3] Lee Breslau, Pei Cao, Li Fan, Graham Phillips, and Scott Shenker, "Web caching and zipf-like distributions: Evidence and implications," in *Proc. of IEEE INFOCOM*, 1999.

[4] Kailun Wang, Na Deng, and Xuanheng Li, "An efficient content popularity prediction of privacy preserving based on federated learning and wasserstein gan," *IEEE Internet of Things Journal*, 2022.

[5] Yanxiang Jiang, Yuting Wu, Fuchun Zheng, Mehdi Bennis, and Xiaohu You, "Federated learning-based content popularity prediction in fog radio access networks," *IEEE Transactions on Wireless Communications*, vol. 21, pp. 3836–3849, 2022.

[6] Yu Xiong, Hao Jin, Tao Feng, Rui Lin Jia, Qing Zhang, and Chen Xi Zhao, "Content popularity prediction based on integrated features and federated learning," *2021 7th IEEE International Conference on Network Intelligence and Digital Content (IC-NIDC)*, pp. 398–402, 2021.

[7] Russell Hardin and Garrett Cullity, "The free rider problem," *Stanford Encyclopedia of Philosophy*, 2003.

[8] William H Press and Freeman J Dyson, "Iterated prisoner's dilemma contains strategies that dominate any evolutionary opponent," *Proceedings of the National Academy of Sciences*, vol. 109, no. 26, pp. 10409–10413, 2012.

[9] Biheng Zhou, Zhihai Rong, and Xiang Yu, "Equalizing payoffs of a structured population in repeated prisoner's dilemma game," *Chaos, Solitons & Fractals*, vol. 192, pp. 116024, 2025.

[10] Ning Zhang, Qian Ma, and Xu Chen, "Enabling long-term cooperation in cross-silo federated learning: A repeated game perspective," *IEEE Transactions on Mobile Computing*, 2022.

[11] Ming Tang and Vincent WS Wong, "An incentive mechanism for cross-silo federated learning: A public goods perspective," in *Proc. of IEEE INFOCOM*, 2021.

[12] Qin Hu, Shengling Wang, Peizi Ma, Xiuzhen Cheng, Weifeng Lv, and Rongfang Bie, "Quality control in crowdsourcing using sequential zero-determinant strategies," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 5, pp. 998–1009, 2019.

[13] Qin Hu, Shengling Wang, Xiuzhen Cheng, Liran Ma, and Rongfang Bie, "Solving the crowdsourcing dilemma using the zero-determinant strategies," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1778–1789, 2019.

[14] Masahiko Ueda, "On the implementation of zero-determinant strategies in repeated games," *Applied Mathematics and Computation*, vol. 489, pp. 129179, 2025.

[15] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang, "On the convergence of fedavg on non-iid data," in *Proc. of ICLR*, 2020.

[16] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al., "Advances and open problems in federated learning," *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021.

[17] Jianan Chen, Qin Hu, and Honglu Jiang, "Social welfare maximization in cross-silo federated learning," in *Proc. of IEEE ICASSP*, 2022.

[18] Qin Hu, Shengling Wang, Rongfang Bie, and Xiuzhen Cheng, "Social welfare control in mobile crowdsensing using zero-determinant strategy," *Sensors*, vol. 17, no. 5, pp. 1012, 2017.