



Article Cluster Content Caching: A Deep Reinforcement Learning Approach to Improve Energy Efficiency in Cell-Free Massive Multiple-Input Multiple-Output Networks

Fangqing Tan ¹,*^(D), Yuan Peng ¹ and Qiang Liu ²,*

- ¹ Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin University of Electronic Technology, Guilin 541004, China; pengyuan@mails.guet.edu.cn
- ² College of Electronic and Information Engineering, Shandong University of Science and Technology, Qingdao 266590, China
- * Correspondence: tfqing@guet.edu.cn (F.T.); qiangliu@sdust.edu.cn (Q.L.)

Abstract: With the explosive growth of micro-video applications, the transmission burden of fronthaul and backhaul links is increasing, and meanwhile, a lot of energy consumption is also generated. For reducing energy consumption and transmission delay burden, we propose a cell-free massive multiple-input multiple-output (CF-mMIMO) system in which the cache on the access point (AP) is used to reduce the load on the link. In this paper, a total energy efficiency (EE) model of a cache-assisted CF-mMIMO system is established. When optimizing EE, forming the co-operation cluster is critical. Therefore, we propose an energy-efficient joint design of content caching, AP clustering, and low-resolution digital-to-analog converter (DAC) in a cache-assisted CF-mMIMO network based on deep reinforcement learning. This scheme can effectively cache content in APs and select the appropriate DAC resolution. Then, taking into account the channel state information and user equipment (UE)'s content request preference, a deep deterministic policy gradient algorithm is used to jointly optimize the cache strategy, AP clustering, and DAC resolution decisions. Simulation results show that the energy efficiency of the proposed scheme is 4% higher than that of other schemes without the resolution optimization and is much higher than that of the only AP clustering without the joint design of content caching and channel quality.

Keywords: content caching; cell-free massive MIMO; energy efficiency; deep deterministic policy gradient algorithm

1. Introduction

Due to the rapid development of smart devices such as smart phones, smart watches, smart robots, and drones, mobile data traffic on wireless networks has experienced terrible growth. IDC estimates that by 2023, there will be 48.9 billion connected devices worldwide [1]. Such a large number of devices will not only generate exabytes of data but will also require massive amounts of content, which will create unprecedented challenges in the upcoming communications field. The capacity of a backhaul link has become the bottleneck of a data-intensive network, and it is necessary to find an efficient way to reduce the backhaul link load to meet the rapidly growing demand for mobile communication.

Caching is a well-known technique used to improve the performance of numerous wired networks, such as content-centric networks [2–4]. In cellular networks, caching frequently requesting content at the edge of the network can reduce backhaul costs, reduce access latency and power consumption, and increase throughput. In [5], it is proposed to replace the backhaul link by caching on base stations (BSs). By optimizing the cache strategy, it is possible to serve more users within the limits of download time, which significantly increases throughput. In [6], the caching in the BS can lighten the backhaul traffic load. In order to minimize the overall energy consumption attributed to cache and



Citation: Tan, F.; Peng, Y.; Liu, Q. Cluster Content Caching: A Deep Reinforcement Learning Approach to Improve Energy Efficiency in Cell-Free Massive Multiple-Input Multiple-Output Networks. *Sensors* 2023, 23, 8295. https://doi.org/ 10.3390/s23198295

Academic Editor: Davy P. Gaillot

Received: 4 September 2023 Revised: 1 October 2023 Accepted: 3 October 2023 Published: 7 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). data transmission including inter-BS and BS-to-server communications, Ref. [7] optimizes the allocation of cache sizes for BSs and service gateways. With the goal of reducing the overall energy consumption of the service, the caching strategy is fine-tuned in [8], where the influence of multicast transmission is considered.

At the same time, to cater to the rising need for traffic data, more and more antennas of base stations and smaller and smaller cell radius will inevitably cause more and more inter-cells interference. To tackle this issue, many research efforts have been made to diminish inter-cell interference [9,10]. Two primary approaches exist: massive multiple-input multiple-output (mMIMO) systems [11] and distributed systems [12]. Within mMIMO systems, BS utilizes the spatial multiplexing yielded by an extensive array of antennas. These antennas are coupled with precoding techniques that can substantially and efficiently mitigate both intra-cell and inter-cell interference among user equipment (UEs). Nonetheless, the uniformity of service quality across all terminals is not guaranteed by the system. The terminal near the BS can enjoy better service due to good channel conditions, while the terminal located at the edge of the cell can only get inadequate quality of service. Within a distributed system, multiple BSs or access points (APs) collaborate by exchanging service data and channel state information (CSI), aiming to minimize inter-cell interference. But distributed systems remain centered around individual cells. Multi-cell collaboration essentially extends the coverage of a single cell. Edge effects continue to impact UE positioned at the periphery of the cell. Therefore, the cell-free massive multiple-input multiple-output (CF-mMIMO) technique has been introduced [13,14], which combines the strengths the aforementioned two systems, namely robust interference cancellation and macro diversity gain. Additionally, it has made some enhancements: (1) It shifts from a cell-centric service model to a UE-centric service model, allowing for potential overlap between distinct AP clusters. (2) There exists a substantial number of APs, wide coverage, and APs are closer to the terminal. Thus, it completely eliminates the concept of a cell.

The essence of the CF-mMIMO system is that the mMIMO system moves the AP closer to the terminals through the integration of fronthaul links and a more frequent utilization of the backhaul links. This will cause a sharp increase in the link load in the CF-mMIMO system, which inevitably results in elevated energy consumption. Therefore, traffic congestion in the fronthaul/backhaul link and high transmission energy consumption constitute the bottlenecks that impede the practical implementation of CF-mMIMO systems. The content caching technology proactively stores data in the cache device and directly transmits the data to the terminal during peak hours without the need to obtain data from the central processing unit (CPU) and core network via the fronthaul/backhaul link. Because the requested content is concentrated in a limited number of popular files [15], the cost associated with its caching is diminishing [16], so content caching proves to be a cost-effective and effective technique in lessening the burden on the link. Building upon this notion, a cache-assisted CF-mMIMO system is introduced in [17]. Moreover, we proposed energy-efficient content of a data caching strategy in CF-mMIMO systems in [18], but only research ideas were provided without experimental validation. Therefore, its total energy efficiency (EE) maximization problem is non-deterministic polynomial-hard (NP-hard) and necessitates solutions through inefficient and non-scalable methods. Additionally, researchers have started to consider the joint optimization of user association and caching strategies [19–21]. For example, in [19], the high-density satellite-UAV-terrestrial network scenario is considered, and the initial combination optimization problem is effectively solved using game theory and genetic algorithm for clustering and cache placement, respectively. In [20], for a CF-mMIMO-assisted vehicle edge network, a Deep-Q-Network (DQN) algorithm was proposed to optimize the cache decision for improving the network capacity and content delivery performance. Moreover, two deep reinforcement learning (DRL) methods, single-agent reinforcement learning and multi-agent reinforcement learning, were proposed to solve the joint optimization problem of user association and content cache in CF-mMIMO in [21]. However, most existing research focuses on content caching strategies in edge caching, without considering AP clustering strategies.

On the other hand, a substantial quantity of modules of high-resolution analog-todigital converters (ADCs) and digital-to-analog converters (DACs) generate a lot of power consumption. To avoid this, it is recommended to use low-resolution ADCs (1–3 bits) in CF-mMIMO networks. This trade-off reduces power consumption while sacrificing spectral efficiency (SE). The work of [22] shows that low-resolution ADCs have better EE than high-resolution ADCs in the uplink of CF-mMIMO systems.

Creating a practical model for the total EE of a cache-assisted CF-mMIMO system, one that is both straightforward to calculate and analyze, while also being amenable to effective optimization, poses a significant challenge. To date, little research has been conducted on cache-assisted CF-mMIMO systems with cache assistance, which encourages the development of this study. The primary contributions of this paper can be outlined as follows:

- In this paper, a new total EE model of a cache-assisted CF-mMIMO system is established, which has the following advantages: the introduction of low-resolution DAC can improve EE; UE-centric cache deployment can provide a better user experience; and considering the influence of different resolution converters on EE, it is more suitable for practical use;
- A deep deterministic policy gradient (DDPG) algorithm is proposed to solve the joint optimization problem of content cache, AP clustering, and DAC resolution, and it can find the global optimal decision for maximizing the EE performance in cache-assisted CF-mMIMO networks;
- We compare and discuss the influence of DAC resolutions, the numbers of UEs, and APs on the EE performance. Moreover, the proposed DDPG method is compared with the benchmark methods, such as clustering based on signal-to-interference noise ratio (SINR) and caching strategies based on content popularity. By exploiting the intelligent design, its EE is not only significantly better than the benchmark (BM) methods but also better than the DDPG method based on the joint content cache and AP clustering.

The rest of the paper is organized as follows. In Section 2, we give a model for the cache-assisted CF-mMIMO system. In Section 3, we propose the total EE model of the cache-assisted CF-mMIMO system and formulate the optimization problem. In Section 4, we present an approach based on DRL. The simulation results and discussion are provided in Section 5. Conclusively, we summarize this paper in Section 6.

2. System Model

In this section, the signal model, cache model, and DAC resolution model of the cache-assisted CF-mMIMO network are introduced. For the signal model, it describes the transmitted signal in the cache-assisted CF-mMIMO network's downlink channel. For the cache model, a content cache mechanism is outlined to enhance the network's EE. For the low-resolution DAC model, the power consumption generated with different resolutions and the effect on the signal transmission are explained.

2.1. Signal Model

Figure 1 depicts an example topology of a dynamic collaborative cluster serving UE in a cache-assisted CF-mMIMO network. We consider a downlink CF-mMIMO, i.e., a cache-assisted CF-mMIMO system encompassing of M single-antenna AP and K single-antenna UE. Every AP is linked to the CPU via a fronthaul link, while the CPU itself connects to the core network via backhaul links. All APs and UEs are distributed randomly across S_a area. And we only focus on downlink transmissions in this paper.



Figure 1. Caching-assisted cell-free massive MIMO model.

In time-division duplex (TDD) mode, all APs provide identical time/frequency resources to each terminal. Let the channel linking the *m*-th AP and the *k*-th UE be

$$g_{mk} = (d_{mk}/d_0)^{-\alpha} h_{mk}$$
(1)

where d_{mk} denotes the distance between the *m*-th AP and the *k*-th UE, $d_0 = \min_{m,k} d_{mk}$ is the reference distance, α represents the path-loss exponent ($\alpha \ge 2$), and $h_{mk} \sim C\mathcal{N}(0,1)$ denotes small-scale fading.

Let \Re_k denote the set of APs serving the *k*-th UE and \mathbb{C}_m represent the set of UE served by the *m*-th AP. We make the assumption that each UE is ensured service from no more than L(L < M) APs (i.e., $|\Re_k| \le L, \forall k$). Therefore, the AP set for all services and the UE set for all services can be represented as $\Re = \bigcup_{k=1}^K \Re_k$ and $\mathbb{C} = \bigcup_{m=1}^M \mathbb{C}_m$, respectively. Let q_k be a symbol emitted in the service AP for the *k*-th UE, where $E[|q_k|^2] = 1, E[q_k] = 0, \forall k$ and $E[q_kq_l^*] = 0, \forall k \ne l$ (i.e., the symbols of distinct UE are not related). Then, the transmitted signal of the *m*-th AP can be expressed as [14]

$$x_m = \sum_{k \in \mathbb{C}_m} \sqrt{p_{mk}} \hat{g}_{mk}^* q_k \tag{2}$$

where p_{mk} signifies the power assigned to the *k*-th UE at the *m*-th AP subject to power constraints, $E[|x_m|^2] \le P_m$ is constrained by the maximum power P_m transmitted by the *m*-th AP, and \hat{g}_{mk} denotes the channel estimation g_{mk} at the *m*-th AP. This paper considers perfect CSI (i.e., $\hat{g}_{mk} = g_{mk}, \forall m, k$).

Accordingly, the *k*-th UE's received signal can be expressed as [23]

$$r_{k} = \sum_{\substack{m \in \Re_{k} \\ g_{mk}x_{m} + w_{k} \\ g_{mk}x_{m} + \sum_{\substack{m \in \Re_{k}^{c} \\ g_{mk}x_{m} + \sum_{\substack{m \in \Re_{k}^{c} \\ g_{mk}x_{m} + w_{k} \\ g_{mk}x_{m} + w_{k} \\ g_{mk}x_{m} + \sum_{\substack{m \in \Re_{k}^{c} \\ g_{mk}x_{m} + w_{k} \\ g_{mk}x_{m} + w_{k} \\ g_{mk}x_{m} + \sum_{\substack{m \in \Re_{k}^{c} \\ k' \in \mathbb{C}, k' \neq k} \sqrt{p_{mk'}}g_{mk}\hat{g}_{mk'}^{*}q_{k'} + w_{k}$$
(3)

where $w_k \sim C\mathcal{N}(0, \sigma_w^2)$ signifies the noise at the *k*-th UE, and $\Re_k^c = \Re \setminus \Re_k$ is the other AP set that does not serve the *k*-th UE.

2.2. Caching Model

We consider a limited file library $CF = \{cf_1, cf_2, \dots, cf_F\}$ with F content files. Let $CF_m \subset CF$ be the set of content files cached in the m-th AP. Additionally, we make the assumption that each AP has the capacity to cache a maximum of N(N < F) files, denoted as $|CF_m| \leq N, \forall m$. Each UE makes content file requests independently or abandons the request. The content file for the k-th UE request is represented by $cf_k \in CF$, where cf_k is determined by the content preference vector for the k-th UE (arranged in descending order of preference for all content files) and the distribution of content preparative (specified with the Zipf distribution). To be more specific, in the content preference vector of the k-th UE, the probability that cf_k equals the content file of the i-th rank is $i^{-\beta}/\sum_{j=1}^F j^{-\beta}$, where β is the Zipf factor; Usually, set to $\beta = 0.5, 1, 2$. Each UE possesses a different, independent, and time-invariant content preference vector.

We use H_{mk} to define the event that the content file requested by the *k*-th UE is cached on its *m*-th service AP, i.e., $cf_k \in CF_m$, $m \in \Re_k$. Therefore, the matching event of the *k*-th UE H_k indicates that the file requested by the *k*-th UE is cached across all APs serving the *k*-th UE, i.e., $cf_k \in CF_m$, $\forall m \in \Re_k$. In case of a miss, there exist certain $m \in \Re_k$ APs that do not cache the file of the *k*-th UE, i.e., $cf_k \notin CF_m$, $\exists m \in \Re_k$. In such scenarios, these APs will necessitate requesting the content file cf_k from the CPU/core network for joint AP transfer. The network's hit ratio is denoted as $H = \sum_{k \in \mathbb{C}} 1_{H_k} / |\mathbb{C}|$, where 1_{H_k} signifies the indicator function, and it is set to be 1 if the H_k event occurs; otherwise, it is set to 0.

2.3. Low-Resolution DAC Model

We adopt a low-resolution DAC with a binary-weighted current-oriented topology, whose power consumption is composed of both static and dynamic components. The power consumption of a DAC module with a resolution of *b* can be given as [24,25]

$$P_{DAC}(b, F_s) = 1.5 \times 10^{-5} \cdot 2^b + 4.5 \times 10^{-12} \cdot b \cdot F_s \tag{4}$$

where F_s is the sampling frequency.

Each AP's antenna is connected to a low-resolution DAC, and the resulting signal has $\alpha \in [0, 1]$ linear gain. Therefore, the transmitting signal of the *m*-th AP given in (2) is now modified as

$$x_m = \alpha_m \sum_{k \in \mathbb{C}_m} \sqrt{p_{mk}} \hat{g}_{mk}^* q_k \tag{5}$$

where α_m represents the linear gain of the *m*-th AP, and its expression is [22,26]

$$\alpha_m = \begin{cases} 0.6366 , b_m = 1 \\ 0.8825 , b_m = 2 \\ 1 - (\pi\sqrt{3}/2) \cdot 2^{-2b_m} , b_m \ge 3 \end{cases}$$
(6)

The received signal of *k*-th terminal provided by (3) is rewritten as

$$r_{k} = \underbrace{\sum_{m \in \Re_{k}} \alpha_{m} \sqrt{p_{mk}} g_{mk} \hat{g}_{mk}^{*} q_{k}}_{\text{useful signal}} + \underbrace{\sum_{m \in \Re_{k'}} \alpha_{m} \sum_{k' \in \mathbb{C}, k' \neq k} \sqrt{p_{mk'}} g_{mk} \hat{g}_{mk'}^{*} q_{k'} + w_{k}}_{\text{interference plus noise}}$$
(7)

3. The EE Model and Problem Formulation

3.1. The System Sum Rate

According to the Shannon theory, the achievable rate of the *k*-th UE can be expressed as

$$R_{k} = B \log_{2} \left(1 + \frac{\left| \sum_{m \in \Re_{k}} \alpha_{m} \sqrt{p_{mk}} g_{mk} \hat{g}_{mk}^{*} \right|^{2}}{\sum_{k' \in \mathbb{C}, k' \neq k} \left| \sum_{m \in \Re} \alpha_{m} \sqrt{p_{mk'}} g_{mk} \hat{g}_{mk'}^{*} \right|^{2} + \left| w_{k} \right|^{2}} \right)$$
(8)

where *B* is the bandwidth. Therefore, the overall achievable rate of the considered cacheassisted CF-mMIMO network is given using

$$R_{sum} = \sum_{k \in \mathbb{C}} R_k \tag{9}$$

3.2. Power Consumption

The overall power consumption of the network consists of four parts: (1) the transmission power of all service APs; (2) power consumption of DAC in all service APs; (3) the power required by the AP to recover the lost content file from the CPU; and (4) the power needed by the CPU to recover the lost content from the core network.

For (1), the total transmitted power of all service APs is represented by $\sum_{m \in \Re} P_m$. For (2), the sum of DAC power consumption in all service APs is given using $\sum_{m \in \Re} P_{DAC}^m$, where P_{DAC}^m indicates the power consumption of the DAC module with b_m resolution selected using the *m*-th AP.

For (3) and (4) in cache-assisted CF-mMIMO systems, all APs within a cluster must concurrently transmit identical content to the terminal. Cache deployment results in three scenarios are illustrated in Figure 2: (a) Every AP within the cluster has deployed the required content. (b) Only some APs have deployed the required data. c) None of the APs deployed the required content. In Scenario (a), no content is conveyed through either the fronthaul or backhaul link. In Scenario (b), certain APs are required to transmit content via the fronthaul link. In Scenario (c), all content is transmitted to the AP via the backhaul link of the core network and the fronthaul link of the CPU.



Figure 2. Three scenarios in which content is requested during transmission.Green arrows represent the access link between APs and UEs, red arrows represent the backhaul/fronthaul links between the core network and CPU or APs and CPU.

The fronthaul link is utilized for content transmission between the AP and the CPU, its power consumption is proportional to the cumulative SE sum, and its expression is [27]

$$P_{bh,m} = E_{bh} \sum_{k \in \mathbb{C}_m} R_k \tag{10}$$

where E_{bh} indicates the energy consumed for transmitting 1 Mbit of data over the fronthaul link.

The *m*-th AP is used to transmit data $q_1, q_2, \dots q_K$ via a fronthaul/backhaul link between the CPU and core network. Therefore, the fronthaul/backhaul power consumption depends on the SE, SE₁, SE₂, \dots SE_{*K*}. If the *m*-th AP serves only specific UEs, it merely transmits data related to these UEs. Therefore, the power consumption for the fronthaul/backhaul is contingent solely on the SE of these UEs. As shown in Figure 2, the cache power consumption is calculated with the user as the center, so the fronthaul power consumption of the *k*-th cluster can be represented as

$$P_{bh,k} = E_{bh} R_k \tag{11}$$

Similarly, the backhaul power generated by the *k*-th cluster's backhaul link for transferring data between the core network and the CPU can be expressed as

$$P_{bb,k} = E_{bb} R_k \tag{12}$$

where E_{bb} indicates the energy consumed for transmitting 1 Mbit of data over a backlink. Therefore, the power uploaded via the AP to the CPU in the *k*-th cluster can be represented as

1

$$P_{bh,k}^{up} = \begin{cases} P_{bh,k} &, 0 < \sum_{m \in \Re_k} \left| H_{mk}^{miss} \right| / \left| \Re_k \right| < 1 \\ 0 &, \text{ others} \end{cases}$$
(13)

where $H_{mk}^{miss} = |1 - H_{mk}|$ represents the event that the UE content request provided by the *m*-th AP is not cached on the *m*-th AP. The value is 1 if the event H_{mk}^{miss} occurs, but it is 0 otherwise.

The energy consumption associated with the content requested by the AP from the CPU in the *k*-th cluster can be expressed as

$$P_{bh,k}^{down} = P_{bh,k} \sum_{m \in \Re_k} \left| H_{mk}^{miss} \right| \tag{14}$$

The backhaul power generated by the *k*-th cluster requesting content from the core network can be denoted as

$$P_{bb,k}^{down} = P_{bb,k} \left[\sum_{m \in \Re_k} \left| H_{mk}^{miss} \right| / \left| \Re_k \right| \right]$$
(15)

Therefore, the fronthaul/backhaul power consumption of the *k*-th cluster can be expressed as

$$P_{B,k} = P_{bh,k}^{up} + P_{bh,k}^{down} + P_{bb,k}^{down}$$

$$\tag{16}$$

So, the overall energy consumption can be expressed as

$$P_{total} = \sum_{m \in \Re} (P_m + P_{DAC}^m) + \sum_{k=1}^{K} P_{B,k}$$
(17)

8 of 17

3.3. Problem Formulation

Our aim is to find a strategy that determines the AP clustering with $\Re_1, \Re_2, \dots, \Re_K$, as well as the AP's content cache CF_1, CF_2, \dots, CF_M and its DAC resolution b_1, b_2, \dots, b_M , in order to maximize the system's EE. The optimization problem can be expressed as

$$\max \frac{R_{sum}}{P_{total}}$$
s.t. (C1): $|\Re_k| \le M, \forall k \in \mathbb{C},$
(C2): $|CF_m| \le L, \forall m \in \Re,$
(C3): $b_m \in N^+, \forall m \in \Re.$
(18)

where constraint (C1) indicates that the count of APs within the AP cluster of each UE cannot exceed the maximum number of connections *M*. Constraint (C2) requires that the amount of content cached on each AP must not exceed its maximum capacity number *L*. Constraint (C3) means that the resolution of each DAC is a positive integer.

To maximize the EE performance, the design for trade-offs is needed. First, AP clusters based on channel quality and high-resolution DAC can select better channels to get the best SE. In contrast, an AP cluster based entirely on cached content and low-resolution DAC can avoid the energy consumption of the fronthaul/backhaul link, reducing the energy consumption of the DAC module. In addition, in large networks, solving this issue is complicated due to the large number of APs and UEs. To solve this problem, we developed deep reinforcement learning-based content caching, AP clustering, and DAC resolution co-selection strategies, which will be elaborated upon in the Section 4.

4. Deep Reinforcement Learning Method

In this section, we will describe how the DDPG algorithm solves the combined predicament of AP clustering, caching, and selecting DAC resolution. Three basic components (action, state, and reward) are defined in reinforcement learning (RL) problems.

4.1. Action, State, and Reward

In slot *t*, action a_t encompasses the processes of clustering, caching, and selecting resolution. Let $a_{mk,t} \in \{0,1\}$, $a_{mcf,t} \in \{0,1\}$, and $a_{mb,t} \in \{0,1\}$ represent the status of *m*-th AP and *k*-th UE service, the *cf* file cache, and the *b* bit resolution switch, respectively, where "1" indicates that the service or cache is successful or enabled and "0" indicates that the service or cache is not served, there is no cache, or it is disabled. So, action a_t can be defined as

$$a_t \triangleq \left\{ a_t^{cl}, a_t^{ca}, a_t^{res} \right\} \tag{19}$$

The sets $a_t^{cl} = \{a_{mk,t} : m \in M, k \in K\}$, $a_t^{ca} = \{a_{mcf,t} : m \in M, cf \in CF\}$, and $a_t^{res} = \{a_{mb,t} : m \in M, b \in N^+\}$ contain aggregate results representing the *t*-th time slot for clustering, caching, and selecting resolution, respectively.

Similarly, the action a_t uniquely determines the sets CF_m , b_m , \Re_k and \mathbb{C}_m , i.e., $CF_m = \{cf: a_{mcf,t} = 1, cf \in CF\}$, $b_m = \{b: a_{mb,t} = 1, b \in N^+\}$, $\Re_k = \{m: a_{mk,t} = 1, m \in M\}$, and $\mathbb{C}_m = \{k: a_{mk,t} = 1, k \in K\}$.

The state considered in RL should be the set of information that the CPU can collect to compute the reward. In this article, the state of the *t*-th slot is characterized as the collection of channel gain $G_t = \{g_{mk,t} : m \in M, k \in K\}$, the action of the preceding time slot, and the historical record of file requests for each UE. Define the history set of user requests as $e_t = \{e_{kcf,t} : k \in K, cf \in CF\}$, where $e_{kcf,t} = \sum_{t'=1}^t 1_{cf_{k,t'=cf}}$ is the *cf* file download from the *k*-th UE request as of time *t*. So, the state can be denoted as

$$s_t \triangleq \{G_t, a_{t-1}, e_t\} \tag{20}$$

According to the objection function of the optimization problem (18), the reward function of the t-th slot is defined as

$$r(s_t, a_t) \triangleq \frac{R_{sum,t}}{P_{total,t}}$$
(21)

where $R_{sum,t}$ and $P_{total,t}$ are given in (9) and (17), correspondingly, and the extra subscript *t* is given to emphasize the dynamic behavior. It is worth noting that the total achievable rate $R_{sum,t}$ is contingent upon the channel conditions G_t , the clustering outcome a_t^{cl} , and the result a_t^{res} of the selection resolution, while the overall power $P_{total,t}$ is contingent upon the caching result a_t^{res} of the selection resolution.

4.2. Deep Deterministic Policy Gradient Approach

DDPG algorithm utilizes an actor–critic network architecture. Moreover, each network is accompanied by its respective target network, resulting in a total of four networks within the DDPG algorithm, namely, the actor network $\mu(\cdot|\theta^{\mu})$, critic network $Q(\cdot|\theta^{Q})$, target actor network $\mu'(\cdot|\theta^{\mu'})$, and target critic network $Q'(\cdot|\theta^{Q'})$. Each network updates according to its own update rules, maximizing cumulative expected returns. Figure 3 gives the schematic diagram of the DDPG algorithm.



Figure 3. The principle of DDPG algorithm.

The DDPG algorithm is well-suited for multi-task learning, aligning with the objectives of this paper. This algorithm enhances training stability by adopting a deterministic strategy, which means it directly outputs a specific action value instead of a probability distribution. The algorithm is trained using an experience replay buffer to store past experiences and then to randomly sample from it. This approach breaks the data correlations and ensures that the data conform to an independent distribution, thereby reducing parameter update variance and enhancing convergence speed. Additionally, experiences can be reused, resulting in high data utilization. DDPG leverages neural networks to represent policies (actor) and value functions (critic), making it suitable for high-dimensional state spaces and capable of learning from vast amounts of perceptual data. In comparison to the widely used DQN, DDPG is particularly apt for continuous action spaces. Furthermore, employing actor networks can improve training efficiency, and having more target actor networks and target critic networks helps prevent potential overestimation issues present in DQN.

Algorithm 1 primarily revises the parameters of the actor network and critic network. The actor network adjusts the weight θ^{μ} by aiming to maximize the cumulative anticipated reward. The critic network adjusts the weight θ^{Q} by seeking to minimize the discrepancy between the evaluation value and the target value. Regarding the update process of the target network, a soft update method is adopted, which can also be called exponential

average motion. That is, the learning rate (or momentum) τ is introduced, and the weighted average of the previous target network parameters and the current corresponding network parameters are subsequently applied to update the target network. Algorithm 1 summarizes the whole DDPG algorithm process.

Α	lgorit	hm 1	DDPG	Alg	orith	ım F	roced	ure
					,			

4.3. Computational Complexity

In the DDPG algorithm, the input dimension of the neural network is denoted as $Input \triangleq M(3K + N + 1) + KN$, the output dimension is denoted as $Output \triangleq M(K + N + b)$, and the number of model parameters is denoted as $Number \triangleq 5Input(Input + 1) + 9Output(Output + 1) + 10Iuput * Output$, determined by the neural network's layer count and layer size. The experience pool's size is Batch = 128, and it holds states, actions, rewards, and next states, resulting in a complexity of O(Batch * (2K(M + N + b) + K(M + N + 1) + 1) + Number). The decision-making process for actions has a time complexity of O(K + 2M), thus leading to the complexity of O(timeslot * (Number + K + 2M)), where *timeslot* stands for the number of training iterations.

5. Simulation Results

5.1. Simulation Settings

In this section, we conduct a comparison and analysis of (1) the EE performance of the proposed RL method with three different BM strategies (called BM1, BM2, and BM3), (2) the convergence behavior of DDPG algorithm, (3) the effect of DAC resolutions on the EE, (4) the impact of the number of UE-associated AP on the EE, and (5) the influence of UE and AP quantity on the EE. The BM strategies are given as follows:

- BM1: clustering policy based on the SINR (*l* ≤ *L* APs to which the *k*-th UE is connected is the *l* with the highest SINR) and caching policy based on local popularity (in the UE served by the *m*-th AP, the most popular *N* files are cached on the *m*-th AP);
- BM2: clustering strategy based on SINR (same as BM1) and caching strategy based on network popularity (caching the *N* most popular files across all UEs in all APs);
- BM3: cache-based clustering strategy (each UE is connected to *l* ≤ *L* APs, and its cache is the content request that best matches each UE in the previous slot) and network-based caching strategy (same as BM2).

The computational complexity of BM strategies is all equal to O(K(M + N + 1)), with BM2 having the minimum time complexity of $O((M \log_2 M) * (KN \log_2 N))$, BM3 following with $O((KM \log_2 M) * (KN \log_2 N))$, and BM1 having the maximum complexity of $O((M \log_2 M) * (MKN \log_2 N))$. Although the complexity of the BM strategy is lower

than that of DDPG algorithm, the optimization effect of the DDPG algorithm is much better than that of the BM strategy.

We contemplate a situation where APs and UEs are randomly distributed within the region of $S_a = 1 \text{ km}^2$, one AP is located at the reference coordinate (0,0). Both the positions of UEs and APs remain constant during the training phase. We set the number of APs and UEs to be M = 10 and K = 5, correspondingly, the cache size to $|CF_m| \le 2$ for each AP, the number of files to |CF| = 10, and the DAC resolution to $b_m \le 5$. Refer to [14,28–32] for other system settings and parameters, which are summarized in Table 1.

Table 1. The simulation parameters.

Parameters	Value
Bandwidth B	20 MHz
Maximum DL transmit power P_m	1000 mW
Energy consumption of fronthaul link E_{bh}	0.25×10^{-3} Joule/Mbit
Energy consumption of backhaul link E_{bb}	$15E_{bh}$
Thermal noise power per UE σ_w^2	$7.457 imes 10^{-13} m W$
Path-loss exponent α	2
Zipf distribution factor β	1

5.2. Numerical Results Analysis

Figure 4 shows the convergence of RL+DAC and RL in Algorithm 1, where the EE values versus training episodes are demonstrated. The diagram is trained 1000 times and then fused together. When the BM strategy is compared at the 10th episode of training, it can be seen that the EE of RL+DAC and RL proposed after the 26th episode of training is better than that of other BM strategies. The EE of RL+DAC also completely outperformed the RL algorithm after about the 150th episode of training. Note that in Algorithm 1, RL+DAC assumes that each AP employs a distinct ADC resolution, which sacrifices some computational time in exchange for the improved performance. In contrast, RL employs the same ADC resolution for all APs, thereby reducing the algorithm's complexity.



Figure 4. The convergence of Algorithm 1.

Figure 5 illustrates the impact on total EE of changes in the relative positions of APs and UEs over time. It can be easily observed that the total EE of the RL+DAC and RL algorithms are always better than that of other BM strategies. In BM schemes, we find that when each UE is attended to by a single AP at moments 0, 3, 4, 7, 8, 10, and 11, their total EE values are higher than others. When each UE is attended to by three AP at moments 1, 5, 6, and 9, the total EE performance is the highest. What this means is that one UE does

not choose more AP services to get a better EE performance. Furthermore, the increasing of the number of UEs will result in a greater number of UEs being served by the APs. Consequently, this necessitates APs to make trade-offs when selecting DAC resolution with the RL+DAC algorithm, significantly diminishing the SE improvement for UEs. When there is an abundance of UEs, this effect becomes nearly equivalent to the average SE achieved in the case of employing the same low resolution at each AP.



Figure 5. The total EE versus times.

Figure 6 depicts the correlation between the number of UEs and the mean SE, where the number of APs is M = 10. As illustrated in Figure 6, the mean SE of UEs diminishes as the quantity of UEs increases and then tends to be stable. This phenomenon arises due to the escalation in the quantity of UEs, leading to a gradual intensification of inter-UE interference. Ultimately, the average SE will become stable. Furthermore, the upsurge in the number of UEs will result in a greater number of UEs being served by the APs. Consequently, this necessitates APs to make trade-offs when selecting DAC resolution within the RL+DAC algorithm, significantly diminishing the SE improvement for UEs. When there is an abundance of UEs, this effect becomes nearly equivalent to the average SE achieved when each AP in the RL algorithm utilizes the same low resolution.



Figure 6. The relationship between the number of UEs and the average SE.

Figure 7 explores the influence of the number of UEs on the total EE, where the number of APs is M = 10. As observed in Figure 7, the overall trend of the total EE decreases as the quantity of UEs increases and then tends to stabilize. This is because the growth of the number of UEs in the early stage is approximately proportional to the energy consumed by the system, and the existence of inter-UE interference will slow the growth of its sum achievable rate, so its total EE continues to decline. When the number of UEs is large, all APs are already serving UE, and augmenting the number of UEs will not lead to an elevation in the power consumption of AP activities, resulting in a smaller increase in total power consumption, so the total EE will tend to balance. It also indirectly validates the result of Figure 6: the increase in the number of UE does not always guarantee better overall system performance. In other words, the higher the number of UE, the interference between UEs will be particularly significant. Note: the total EE of K = 3 UEs in Figure 7 is lower than the total EE of K = 4. This is due to the different location of AP and UE, which will lead to different channel conditions, so that the total EE will produce a certain range fluctuation when the quantity of UE is determined. When the quantity is smaller, the fluctuation due to the different effects of the location will be greater.



Figure 7. The relationship between the number of UEs and the total EE.

Figure 8 shows the correlation between the quantity of APs and the sum achievable rate, where the number of UEs is set as K = 5. It is readily noticeable that the sum achievable rate firstly increases with the number of APs and then tends to be stable. This is because when the number of APs is small, the UE selects the APs with better channel conditions, so that the rate can be increased. Nevertheless, in scenarios where the number of APs is large, the augmentation for the number of APs brings about a gradual intensification of interference between APs. Consequently, when the number of APs is already relatively high, further increasing the number of APs will not lead to an increase in the sum achievable rate; in fact, it might even decrease it. Moreover, as depicted in Figure 8, the curves for BM1 (l = 3) and BM2 (l = 3) overlap due to their shared clustering policies, distinct caching strategies, and the fact that the sum achievable rate is solely contingent on SINR and not influenced by caching.

Figure 9 shows the impact of the total quantity of AP on the total EE, where the quantity of UE is K = 5. The simulation diagram also indirectly verifies the result of Figure 8, i.e., the more AP, the overall system performance is not necessarily better. At the same time, it illustrated from Figure 9 that when the quantity of APs is 16, the total EE of the system is the highest. Because as the quantity of APs increases, so does their power

consumption, and when their sum achievable rate increases slowly, their total EE will begin to decrease. In addition, note that the sum achievable rate and total EE of 20 for the number of AP in Figures 8 and 9 do not strictly follow the trend. This is caused by fluctuations due to the randomness of the positions of APs and UEs.



Figure 8. The relationship between the number of APs and the sum achievable rate.



Figure 9. The relationship between the number of APs and the total EE.

In Figure 10, the impact of low-resolution DAC on the total EE is demonstrated, and it can be observed that when $b \ge 6$, the total EE decreases as the resolution b increases. This means that resolution b will achieve a better total EE performance in the interval [1,5], and the RL+DAC resolution in the figure is $b_m \le 5$, $\forall m \in \Re$, which has the best total EE. Therefore, this also validates the wisdom of limiting the resolution range to $b \le 5$ in our RL+DAC algorithm design. In addition, with the increasing resolution b, its total EE decreases faster and faster because in Formula (4), part of the DAC module's power

consumption increases exponentially with the increase in resolution b, while in Formulas (6) and (8), with the resolution b > 5, it becomes evident that the sum achievable rate tends to be stable.



Figure 10. The total EE versus the DAC resolutions.

6. Conclusions

In this paper, an innovative and practical total EE model of a cache-assisted CFmMIMO system is established. To maximize the total EE, an energy-efficient joint design of content cache, AP clustering, and low-resolution DAC is carried out, and then, a DRL algorithm (i.e., DDPG method) is proposed. Numerical results show that the total EE of the RL+DAC strategy considering DAC resolution is generally 4% higher than that of the RL strategy, and the total EE of these strategies is much higher than those of the BM strategies. In addition, it can be expected that for multi-antenna APs, the number of DAC modules increases linearly with the increase in the quantity of antennas, so the total EE of our proposed RL+DAC strategy will be much higher than that of the RL strategy.

Author Contributions: Conceptualization, F.T. and Y.P.; methodology, Y.P. and F.T.; software, Y.P.; validation, F.T. and Q.L.; writing—original draft preparation, F.T. and Y.P.; writing—review and editing, Y.P., F.T. and Q.L.; supervision, F.T. and Q.L.; project administration, F.T. and Q.L.; funding acquisition, F.T. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been supported by the National Natural Science Foundation of China under Grant 62261013 and in part by the Director Foundation of Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing under Grant GXKL06220104.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:					
AP	Access Point				
EE	Energy Efficiency				
mMIMO	Massive Multiple-Input Multiple-Output				
CF-mMIMO	Cell-free Massive Multiple-Input Multiple-Output				
DDPG	Deep Deterministic Policy Gradient				
BS	Base Station				
DAC	Digital-to-Analog Converter				
UE	User Equipment				
CPU	Central Processing Unit				
BM	Benchmark				
SE	Spectrum Efficiency				
DRL	Deep Reinforcement Learning				
CSI	Channel Statement Information				

References

- 1. Rydning, J. IDC Worldwide Global DataSphere IoT Device and Data Forecast, 2019–2023; IDC: Needham, MA, USA, 2019; p. 3.
- Choi, N.; Guan, K.; Kilper, D.C.; Atkinson, G. In-network caching effect on optimal energy consumption in content-centric networking. In Proceedings of the IEEE International Conference on Communications, Ottawa, ON, Canada, 10–15 June 2012; pp. 2889–2894.
- Llorca, J.; Tulino, A.M.; Guan, K.; Esteban, J.; Varvello, M.; Choi, N.; Kilper, D.C. Dynamic in-network caching for energy efficient content delivery. In Proceedings of the 32nd IEEE International Conference on Computer Communications, Turin, Italy, 14–19 April 2013; pp. 245–249.
- 4. Li, J.; Liu, B.; Wu, H. Energy-efficient in-network caching for content-centric networking. *IEEE Commun. Lett.* 2013, 17, 797–800. [CrossRef]
- Golrezaei, N.; Shanmugam, K.; Dimakis, A.G.; Molisch, A.F.; Caire, G. FemtoCaching: Wireless video content delivery through distributed caching helpers. Proceedings of The 31st Annual IEEE International Conference on Computer Communications, Orlando, FL, USA, 25–30 March 2012; pp. 1107–1115.
- 6. Bastug, E.; Bennis, M.; Debbah, M. Living on the edge: The role of proactive caching in 5G wireless networks. *IEEE Commun. Mag.* **2014**, *52*, 82–89. [CrossRef]
- Xu, Y.; Li, Y.; Wang, Z.; Lin, T.; Zhang, G.; Ci, S. Coordinated caching model for minimizing energy consumption in radio access network. In Proceedings of the IEEE International Conference on Communications, Sydney, Australia, 10–14 June 2014; pp. 2406–2411.
- Poularakis, K.; Iosifidis, G.; Sourlas, V.; Tassiulas, L. Multicast-aware caching for small cell networks. In Proceedings of the IEEE Wireless Communications and Networking Conference, Istanbul, Turkey, 6–9 April 2014; pp. 2300–2305.
- 9. Rusek, F.; Persson, D.; Lau, B.K.; Larsson, E.G.; Marzetta, T.L.; Edfors, O.; Tufvesson, F. Scaling up MIMO: Opportunities and challenges with very large arrays. *IEEE Signal Process. Mag.* **2013**, *30*, 40–60. [CrossRef]
- Lopezperez, D.; Roche, G.; Kountouris, M.; Quek, T.; Jie, Z. Enhanced inter-cell interference coordination challenges in heterogeneous networks. *IEEE Wirel. Commun.* 2011, 18, 22–30. [CrossRef]
- Larsson, E.G.; Edfors, O.; Tufvesson, F.; Marzetta, T.L. Massive MIMO for next generation wireless systems. *IEEE Commun. Mag.* 2014, 52, 186–195. [CrossRef]
- 12. Gesbert, D.; Hanly, S.; Huang, H.; Shitz, S.S.; Simeone, O.; Yu, W. Multi-cell MIMO cooperative networks: A new look at interference. *IEEE J. Sel. Areas Commun.* 2010, *28*, 1380–1408. [CrossRef]
- 13. Ammar, H.A.; Adve, R.; Shahbazpanahi, S. User-centric cell-free massive MIMO networks: A survey of opportunities, challenges and solutions. *IEEE Commun. Surv. Tutor.* 2022, 24, 611–652. [CrossRef]
- 14. Ngo, H.Q.; Ashikhmin, A.; Yang, H.; Larsson, E.G.; Marzetta, T.L. Cell-free massive MIMO versus small cells. *IEEE Trans. Wirel. Commun.* **2017**, *16*, 1834–1850. [CrossRef]
- 15. Wang, K.; Chen, Z.; Liu, H. Push-based wireless converged networks for massive multimedia content delivery. *IEEE Trans. Wirel. Commun.* **2014**, *13*, 2894–2905.
- 16. Peng, M.; Sun, Y.; Li, X.; Mao, Z.; Wang, C. Recent advances in cloud radio access networks: System architectures, key techniques, and open issues. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 2282–2308. [CrossRef]
- 17. Chen, S.; Zhang, J.; Björnson, E.; Wang, S.; Xing, C.; Ai, B. Wireless caching: Cell-free versus small cells. In Proceedings of the IEEE/CIC International Conference on Communications in China, Xiamen City, China, 28–30 July 2021; pp. 1–6.
- Peng, Y.; Tan, F.; Liu, Q. Energy-efficient content caching strategy in cell-free massive MIMO networks with reinforcement learning. In Proceedings of the 2023 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Beijing, China, 14–16 June 2023; pp. 1–3.

- 19. Nguyen, M.-H.T.; Bui, T.T.; Nguyen, L.D. Real-time optimized clustering and caching for 6G satellite-UAV-terrestrial networks. *IEEE Trans. Intell. Transp. Syst.* 2023, 1–11. [CrossRef]
- Chaowei, W.; Ziye, W.; Lexi, X. Collaborative Caching in Vehicular Edge Network Assisted by Cell-Free Massive MIMO. *Chin. J. Electron.* 2023, 33, 1–13.
- Chuang, Y.-C.; Chiu, W.-Y.; Chang, R.Y.; Lai, Y.-C. Deep reinforcement learning for energy efficiency maximization in cacheenabled cell-free massive MIMO networks: Single- and multi-agent approaches. *IEEE Trans. Veh. Technol.* 2023, 72, 10826–10839. [CrossRef]
- Zhang, Y.; Zhou, M.; Qiao, X.; Cao, H.; Yang, L. On the performance of cell-free massive MIMO with low-resolution ADCs. *IEEE Access* 2019, 7, 117968–117977. [CrossRef]
- Chang, R.Y.; Han, S.-F.; Chien, F.-T. Reinforcement learning based joint cooperation clustering and content caching in cell-free massive MIMO networks. In Proceedings of the 2021 IEEE 94th Vehicular Technology Conference (VTC2021-Fall), Norman, OK, USA, 27–30 September 2021; pp. 1–7.
- 24. Cui, S.; Goldsmith, A.J.; Bahai, A. Energy-constrained modulation optimization. IEEE Trans. Wirel. Commun. 2005, 4, 2349–2360.
- Ribeiro, L.N.; Schwarz, S.; Rupp, M.; de Almeida, A.L.F. Energy efficiency of mmWave massive MIMO precoding with lowresolution DACs. *IEEE J. Sel. Top. Signal Process.* 2018, 12, 298–312. [CrossRef]
- Zhang, J.; Dai, L.; He, Z.; Jin, S.; Li, X. Performance analysis of mixed-ADC massive MIMO systems over Rician fading channels. *IEEE Trans. Wirel. Commun.* 2017, 35, 1327–1338. [CrossRef]
- Ngo, H.Q.; Tran, L.-N.; Duong, T.Q.; Matthaiou, M.; Larsson, E.G. On the total energy efficiency of cell-free massive MIMO. *IEEE Trans. Green Commun. Netw.* 2018, 2, 25–39. [CrossRef]
- Sadeghi, A.; Sheikholeslami, F.; Giannakis, G.B. Optimal and scalable caching for 5G using reinforcement learning of space-time popularities. *IEEE J. Sel. Top. Signal Process.* 2018, 12, 180–190. [CrossRef]
- 29. Yang, C.; Yao, Y.; Chen, Z.; Xia, B. Analysis on cache-enabled wireless heterogeneous networks. *IEEE Trans. Wirel. Commun.* 2016, 15, 131–145. [CrossRef]
- Zhong, C.; Gursoy, M.C.; Velipasalar, S. Deep multi-agent reinforcement learning based cooperative edge caching in wireless networks. In Proceedings of the IEEE International Conference on Communications, Shanghai, China, 20–24 May 2019; pp. 1–6.
- 31. Björnson, E.; Sanguinetti, L. Scalable cell-free massive MIMO systems. IEEE Trans. Commun. 2020, 68, 4247–4261. [CrossRef]
- 32. Zhang, H.; Li, H.; Liu, T.; Dong, L.; Shi, G.; Gao, X. Lower energy consumption in cache-aided cell-free massive MIMO systems. *Digit. Signal Process.* **2023**, 135, 103936. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.