2023 19th International Conference on Mobility, Sensing and Networking (MSN) | 979-8-3503-5826-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/MSN60784.2023.00098

Auction-Based Dynamic Resource Allocation in Social Metaverse

Nan Liu†, Tom H. Luan†, Yuntao Wang†, Yiliang Liu†, and Zhou Su†*

†School of Cyber Science and Engineering, Xi'an Jiaotong University, Xi'an, China
*Corresponding Author: zhousu@ieee.org

***Abstract*—The emergence of the Metaverse has brought forth a new era of social networks, offering immersive virtual spaces for users to engage in social activities. However, the resource-intensive nature of rendering avatars and virtual scenes places considerable strain on end devices. To improve the Quality of Experience (QoE) for users, the utilization of edge servers' resources becomes crucial. Moreover, accommodating the diverse QoE requirements and time dynamics of users (*e.g.*, user join/departure, and social activities) escalates the complexity of resource allocation. In this paper, we propose an auction-based dynamic resource allocation algorithm to efficiently and economically allocate various limited resources (*e.g.*, CPU, GPU, RAM, and VRAM) of edge servers to social user groups in a rapid and decentralized manner. First, with heterogeneous and dynamic varying resources at each Planet (*i.e.*, edge server to host Metaverse users), we design an optimal Planet access scheme to help social user groups to determine which Planet to connect. Second, considering the dynamic nature of social applications, *e.g.*, users dynamically join and depart the network with dynamic requirements on resources, we present a multi-round auction game between social user groups and edge servers to compete for the dynamic multi-dimensional resources before each scheduled time period. By using the above mechanisms, our scheme optimizes the dynamic resource utilization by considering the social feature of Metaverse. Using extensive simulations, we demonstrate that the proposed algorithm dynamically and effectively allocates resources for social Metaverse activities, outperforming conventional allocation approaches.**

***Index Terms*—Social Metaverse, dynamic resource allocation, auction game**

I. Introduction

The Metaverse constructs a three-dimensional virtual shared space that can enable users to engage in immersive social activities, offering a revolutionary model for future social networks [1]–[3]. The existing applications of the Metaverse, such as virtual reality (VR) games, Metaverse social platforms, and virtual music concerts, demonstrate its immense value and potential, and have garnered considerable interest from both industry and academia. Notably, in 2021, Facebook, a well-known social media giant, rebranded itself as "Meta" [2], signifying its official transition into a social Metaverse company. Moreover, the virtual concert featuring Travis Scott in the online game Fortnite attracted more than 12 million players [4], an unprecedented number for a live concert.

A wave of research on Metaverse has been sparked with diverse topics focusing on VR presentation, artificial intelligence, blockchain, and networking [5]. With the miniaturization of sensors, the maturity of embedded technology and VR technology, head-mounted displays and other VR devices are expected to become primary terminals for users to engage in social activities in Metaverse [6]. Artificial intelligence is known as the "brain" of Metaverse [7], which provides users with personalized services in social activities through big data inference and learning from extensive multimodal inputs [8], [9]. The virtual economy is a crucial part of the Metaverse community, and blockchain technology offers an open and decentralized solution for constructing a sustainable virtual economy [10], [11]. Networking technologies such as software-defined networking and network slicing allow flexible, scalable management of large-scale Metaverse networks and real-time transmission of vast data [5], [12].

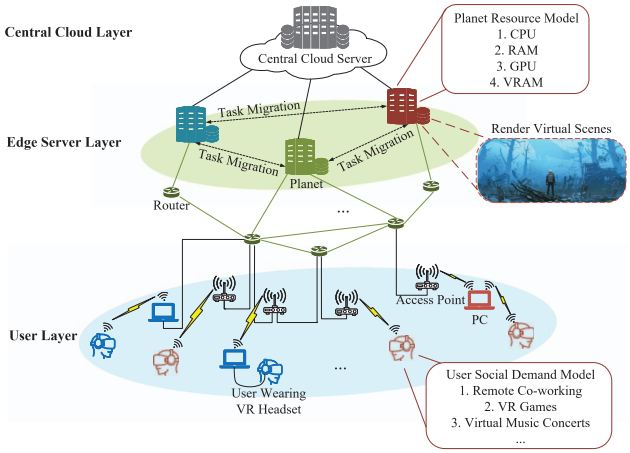

Fig. 1. Three-layer network architecture of Metaverse.

Fig. 1 illustrates the basic network architecture of a Metaverse system which consists of three layers. The top layer is a central cloud server that stores users' data, such as virtual assets and identities. The middle layer comprises edge servers, referred to as *Planets*, located in different regions of Internet. These servers render virtual scenes and provide personalized spaces for users. The bottom layer consists of Metaverse users who connect to Metaverse applications through VR headsets (or VR headsets empowered by personal computers). Users from different locations can participate in Metaverse social activities by logging into Planets, including popular collaborative activities like VR gaming and remote co-working.

This study focuses on the allocation of Planets' resources

979-8-3503-5826-1/23/$31.00 ©2023 IEEE
DOI 10.1109/MSN60784.2023.00098

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

in Metaverse applications. In particular, we explore the social feature of Metaverse. Specifically, in such an immersive social platform, users of Metaverse often have strong *social connections* with others and tend to form *social groups* to engage in virtual activities [2]. For instance, if users from different regions around the world want to attend the same academic conference in a Metaverse space, they form a user group, and their social interactions cannot be overlooked when designing the system. In the meantime, as the Planet has limited resources to host parallel social activities for groups, efficient resource allocation is crucial. In Metaverse applications, artificial intelligence generated content (AIGC) technology can be used to create various types of contents, including virtual items, avatars, and virtual scenes, which enhance the realism of the Metaverse experience [13], [14]. However, rendering avatars and virtual scenes using AIGC can be resource-consuming in terms of the GPU and CPU [15]. Therefore, to economically and efficiently use the limited resources of different Planets to best serve the different social groups is key to the performance of Metaverse.

However, the resource allocation of Planets faces three challenges. *1) Heterogeneous QoE requirements.* With different computing and AI capabilities, different Planets may offer various levels of QoE. In addition, as users in the same Metaverse social group may come from different regions of the world, they have different connection bandwidth and QoE preferences. The Planet resource allocation should consider the heterogeneity of both Planets and users in the social group. *2) System dynamics.* As social activities in Planets evolve and users can dynamically join and depart, the demand for resources of Planets is time-varying. The resource allocation scheme should adapt to the dynamics of virtual activities and plans for futuristic usage. *3) Distributed algorithm.* As a Planet may host a large-scale and dynamic body of users and social groups, there needs a distributed algorithm to achieve a global resource allocation solution while relying solely on the existing knowledge of Planets and all user groups [16].

To address the aforementioned problems, this paper proposes a dynamic resource allocation scheme using game theory. Specifically, we first divide the time horizon into multiple periods and creates an auction game between user groups for each period. The goal of each user group is to select the best *Planet access scheme* to maximize its payoff, which is the difference between its utility and overall payments, while ensuring that each member has a satisfactory QoE. As a resource provider, each Planet must provide resources to offer social Metaverse services to its users. Each Planet should establish a reasonable *resource pricing scheme* that attracts more users while ensuring that the allocated resources do not exceed its capacity limit.

The remainder of this paper is organized as follows. Sect. II characterizes the system model and formulates the optimization problem. Sect. III describes the proposed distributed dynamic resource allocation algorithm. The numerical simulations are shown in Sect. IV. The paper is concluded in Sect. V.

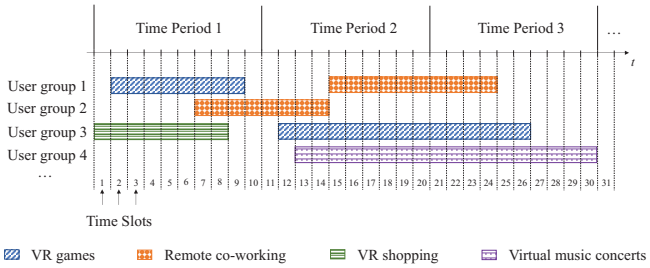

Fig. 2. An illustrating example of dynamic social demand model of 4 user groups under 4 kinds of social activities (*i.e.*, VR games, remote co-working, VR shopping, and virtual music concerts).

II. System Model and Problem Formulation

A. Network Model

The investigated system is illustrated in Fig. 1. We consider a three-layer social Metaverse network architecture consisting of a central cloud server, a set of $\mathcal{J} = \{1, \ldots, J\}$ Planets, and a set of $\mathcal{I} = \{1, \ldots, I\}$ user groups. The central cloud server stores users' basic information such as avatars and virtual assets. Each user group $i \in \mathcal{I}$ has a set of $\mathcal{G}_i = \{1, \ldots, G_i\}$ users who participate in the same social Metaverse activity simultaneously. Users from the same group may be located in different geographical locations, and the network distance between each user $u \in \mathcal{G}_i$ and each Planet j is denoted by $d_{u,j}$, indicating the number of routers along the shortest data transmission path between user u and Planet j. Assuming that there is a set $\mathcal{S}$ of social Metaverse activities which user groups can attend in the system, including remote co-working, VR gaming, VR concerts, etc. Due to limitations of local device resources, user groups need to upload a portion or all of the data generated during social Metaverse activities to Planets for processing. Planets are deployed in different regions close to the users. Each Planet obtains basic information of users accessing it from the central cloud server and provides them with various types of social Metaverse services. Each Planet has a set $\mathcal{R}$ of R types of resources such as GPU, CPU, RAM and VRAM. We denote the capacity vector $\mathbf{C}_j = [C_{j,1}, \ldots, C_{j,R}]$ to represent the amount of available resources owned by Planet j. When processing unit data generated from different activities, a Planet consumes different amounts of each type of resource. For instance, VR gaming, compared to remote co-working, involves more frequent scene changes and requires more realistic scene rendering. Therefore, processing unit data requires more GPU resource in VR gaming than in remote co-working. We denote the consumption vector $\mathbf{c}_s^j = [c_{s,1}^j, \ldots, c_{s,R}^j]$ to represent the resource consumption of Planet j when processing unit data in activity type s.

B. Dynamic Social Demand Model of User Group

Considering that a user group may participate in various social Metaverse activities at different times, the total amount of data generated by all user groups changes dynamically. Consequently, the demand for resources also varies dynamically. Planets need to adjust their resource pricing schemes

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

TABLE I
A SUMMARY OF KEY NOTATIONS

Notation	Description
$x_{i,j}$	The amount of data that user group i plans to upload to Planet j for processing
X_i	The total amount of data generated by user group i
$p_{j,r}$	The price of leasing a unit amount of resource r in Planet j
$c^j_{s,r}$	The resource consumption of Planet j when processing unit data in activity type s
$r_{i,s}$	The proportion of data generated by user group i during activity s to X_i
$d_{u,j}$	The network distance between user u and Planet j
$n_{i,s}$	The payment of group i buying bandwidth resource from a single router for unit data in activity s
$q_{j,r}$	The operational expenditures incurred by Planet j for utilizing unit amount of resource r
$D_{i,s}$	The duration of activity s conducted by user group i during this time period
$A_{i,t,s}$	Whether or not user group i engages in activity s in time slot t
$C_{j,r}$	The total amount of resources r owned by Planet j

accordingly to meet user groups' dynamic demands. In order to design a flexible service provisioning and resource allocation scheme in such a dynamic system, this paper divides the time horizon into multiple time periods, and each time period has Z time slots. We assume that each user group has a plan for the activities to be carried out in the next time period, which includes the types of virtual social activities to be carried out, as well as their start and end times. This information is presented in a timetable denoted as $\mathcal{T}$. For example, the timetable of user group i for activity type $s \in \mathcal{S}$ is denoted by $\mathcal{T}_{i,s} = \{A_{i,s,t} | t = 1, \ldots, Z\}$, where $A_{i,s,t}$ takes values from the set $\{0, 1\}$, and $A_{i,s,t} = 1$ denotes that user group i will engage in activity s in time slot t. Before the start of each time period, each user group determines the Planet access scheme, including which Planet to access and the optimal amount of data to upload, and each Planet determines pricing scheme to be executed during the upcoming time period. We assume that during a time period, user groups do not change their Planet access schemes and Planets do not change their pricing schemes. Some user groups may span multiple time periods when conducting a social Metaverse activity, because Planets have different pricing schemes in different time periods, these user groups need to consider whether to migrate to new Planets before entering the next time period.

Illustrating Example. An example of dynamic social demands model for user groups is shown in Fig. 2. Taking group 1 as an example, it has two social Metaverse activity demands during time periods 1∼3. It engages in VR gaming from time slot 2 to time slot 9 and remote co-working from time slot 15 to time slot 24. Group 1 needs to decide which Planets to access for VR gaming and remote co-working before time period 1 and time period 2 starts, respectively. Specifically, group 1 needs to decide whether the second half of remote co-working will be migrated to another Planet before time slots 21 starts due to the update of the Planets' pricing schemes.

The amount of data generated by a user per unit time slot that requires processing is also dynamic. It is assumed that the amount of data generated by a user within a time slot follows a Gaussian distribution, and its mean and standard deviation are related to the type of social activity involved. For example, if a user is engaged in social activity type s, the mean and standard deviation of the amount of data generated by the user within a time slot are μ_s and σ_s, respectively.

The key notations are summarized in Table I.

C. Payoff Model of User Group

In each time period, the payoff of each user group is the difference between its utility U_i and its payment, which is denoted as follows,

$$\begin{aligned}\Pi_i^{ug} =& U_i(\sum_j x_{i,j}) \\ &- \sum_j (\sum_{s,r} p_{j,r} c^j_{s,r} r_{i,s} + \sum_s n_{i,s} r_{i,s} \sum_{u \in \mathcal{G}_i} d_{u,j}) x_{i,j} \\ &- \sum_j d_{j',j} m_i \beta sgn(x_{i,j}),\end{aligned} \tag{1}$$

where $x_{i,j}$ represents the amount of data that user group i intends to upload to Planet j for processing. Utility U_i is the function of the amount of data to be processed non-locally of group i. Considering that each user group wants to alleviate the data processing burden on its members' terminal devices, we design U_i as a continuously differentiable and strictly concave function that monotonically increases, where its derivative satisfies $U_i^{'}(z) = (X_i/z)^{\alpha_i}, \alpha_i \in [1, 2]$ [17]. X_i represents the total amount of data generated by user group i in this time period.

The second term of Π_i^{ug} represents the payment of uploading data to Planets for processing, which includes the resource leasing cost paid to the Planet and the bandwidth cost paid to the routers. $p_{j,r}$ represents the price of leasing unit amount of resource r in Planet j. $r_{i,s}$ represents the proportion of data generated by user group i during activity s to its total generated data. To ensure that each member in a user group has a good QoE during social Metaverse activities, it is necessary to purchase bandwidth resources from routers along the data transmission path in the network. $n_{i,s}$ represents the cost of user group i purchasing bandwidth resource from a router for unit data in activity s.

The third term of Π_i^{ug} represents the payment of task migration. If a user group engages in an activity that spans multiple time periods and chooses different Planets to access during different time periods, it incurs payment of task migration. j' represents the Planet chosen for access by the user group i in the current time period. m_i represents the amount of data that user group i needs to migrate. β denotes the cost per unit data migration paid by the user group to a single router. $sgn(x_{i,j})$ is the indicator function where $sgn(x_{i,j}) = 1$ when $x_{i,j} > 0$ and $sgn(x_{i,j}) = 0$ when $x_{i,j} = 0$.

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

D. Payoff Model of Planet

Planet utilizes its resources to offer various social Metaverse services that cater to the diverse social activity demands of its users. Using the resources of Planet, user groups should pay the corresponding payments. So in each time period, Planet's revenue is the resource usage payments paid by user groups [18]. In addition, while using resources to provide services to users, Planet needs to consider the corresponding operating expenditures [19]. Therefore, the payoff of each Planet in each time period can be denoted as follows:

$$\Pi_j^{Pt} = \sum_{i,s,r} p_{j,r} c_{s,r}^j r_{i,s} x_{i,j} - \sum_{i,s,r} q_{j,r} c_{s,r}^j r_{i,s} x_{i,j}, \quad (2)$$

where $q_{j,r}$ represents the operational expenditures incurred by Planet j for using one unit of resource r, including electricity costs and so on.

E. Optimization Problem Formulation

For each Planet, it aims to determine a reasonable resource pricing scheme $\mathbf{p}_j = \{p_{j,r} | r = 1, 2, \ldots, R\}$ that satisfies the following three constraints:

- $\Pi_j^{Pt} \geq 0$, so each Planet should ensure that $p_{j,r} \geq q_{j,r}$;
- In each time slot, the amount of resource allocated by each Planet can't exceed the Planet's capacity of that kind of resource, *i.e.*, $\sum_{i,s} x_{i,j} r_{i,s} c_{s,r}^j A_{i,t,s} / D_{i,s} \leq C_{j,r}, \forall j, r, t$;
- When satisfying the above two constraints, the resource prices should be kept as low as possible to attract more users to engage in social Metaverse activities.

After observing the resource prices announced by Planets, each user group aims to determine the optimal Planet access scheme $\mathbf{x}_i = \{x_{i,j} | j = 1, 2, \ldots, J\}$ by solving the following optimization problem.

$$\mathbf{P1}: \max_{\mathbf{x}_i} \quad \Pi_i^{ug} \quad s.t. \quad \begin{cases} C1: \sum_j x_{i,j} \leq X_i, \\ C2: x_{i,j} \geq 0, \ \forall j, \end{cases} \quad (3)$$

where C1 denotes that the amount of data processed by Planets cannot exceed the total data volume X_i.

III. Auction-Based Distributed Dynamic Resource Allocation

In this section, we propose a distributed dynamic resource allocation (DDRA) algorithm by formulating an auction game between user groups and the price-taking Planets. In the auction game for each time period, user groups act as bidders, while Planets act as suppliers, offering R types of divisible resources to facilitate social Metaverse activities for users. In the proposed algorithm, given the initial resource leasing prices set by Planets, each user group determines its Planet access scheme and submits the corresponding bids. Once all bids are received, each Planet updates its resource leasing prices and broadcasts them to all user groups. Upon receiving the updated prices, each user group adjusts its Planet access scheme and submits the corresponding updated bids [20]. This process continues until reaching a stable point, where the resource prices of the Planets become stable and no user group wishes to update their bids. This point is called as the Nash Equilibrium (NE) [21], [22].

Next, we provide the definition of the NE, followed by presenting the optimal Planet access scheme for user group and the resource pricing scheme for Planet. Finally, we outline the DDRA algorithm.

Definition 1. *The auction game is said to be in a Nash Equilibrium if there exists an allocation $\mathbf{x}^*$ and a set of resource prices $\mathbf{p}^* = \mathbf{p}(\mathbf{x}^*)$, such that no user group can unilaterally deviate from its current Planet access scheme to obtain a higher payoff, i.e.,*

$$\Pi_i^{ug}(\mathbf{x}_i^*; \mathbf{x}_{-i}^*; \mathbf{p}^*) \geq \Pi_i^{ug}(\mathbf{x}_i; \mathbf{x}_{-i}^*; \mathbf{p}^*), \forall i, \quad (4)$$

where $\mathbf{x}_i$ represents any Planet access scheme of group i that satisfies $\mathbf{x}_i \neq \mathbf{x}_i^*$ and $\mathbf{x}_{-i}^*$ are optimal strategies of all user groups except for group i.

A. Planet Access Scheme of User Group

To ensure a streamlined user experience, members of a user group choose the same Planet to access. Therefore, the Planet access scheme of a user group involves determining which Planet to connect to and the optimal amount of data to upload. We categorize all user groups into two classes based on whether task migration is required, and discuss the optimal Planet access schemes for each class of user groups. In order to simplify the expression of Π_i^{ug}, we let $f_{i,j} = \sum_{s,r} p_{j,r} c_{s,r}^j r_{i,s} + \sum_s n_{i,s} r_{i,s} \sum_{u \in \mathcal{G}_i} d_{u,j}$. The simplified expression of Π_i^{ug} is shown as follows,

$$\Pi_i^{ug} = U_i(\sum_j x_{i,j}) - \sum_j f_{i,j} x_{i,j} - \sum_j d_{j',j} m_i \beta sgn(x_{i,j}). \quad (5)$$

Then we present the optimal Planet access scheme for each user group in Theorem 1.

Theorem 1. *For user group without task migration requirement, it accesses the Planet with minimum $f_{i,j}$. For user group with task migration requirement, it accesses the Planet which can achieve the maximum Π_i^{ug}. If group i chooses Planet j to access, then the optimal amount of data to upload for processing is shown as follows,*

$$x_{i,j}^* = \begin{cases} X_i, & \text{if } f_{i,j} < U_i'(X_i), \\ U_i'^{(-1)}(f_{i,j}), & \text{if } f_{i,j} \geq U_i'(X_i). \end{cases} \quad (6)$$

Proof of Theorem 1. We consider the two classes of user groups separately.

For user groups without task migration requirements ($m_i = 0$), if they access to Planet j, their payoff is $\Pi_i^{ug} = U_i(x_{i,j}) - f_{i,j} x_{i,j}$. By computing the derivative $\partial \Pi_i^{ug} / \partial x_{i,j}$ and considering the domain of $x_{i,j}$, we obtain the optimal data upload amount for group i if it accesses Planet j as shown in (6), where we see $x_{i,j}^*$ is the function of $f_{i,j}$. Then we observe

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

the impact of $f_{i,j}$ on Π_i^{ug} by substituting $x_{i,j}^*$ into the payoff expression, which is shown as follows,

$$\Pi_i^{ug} = \begin{cases} U_i(X_i) - f_{i,j}X_i, & \text{if } f_{i,j} < U_i'(X_i), \\ U_i(U_i^{'(-1)}(f_{i,j})) - f_{i,j}U_i^{'(-1)}(f_{i,j}), & \text{if } f_{i,j} \geq U_i'(X_i). \end{cases} \tag{7}$$

It is easy to prove that Π_i^{ug} is a monotonically decreasing function of $f_{i,j}$. Therefore, we conclude that the optimal Planet access scheme for each user group who has no task migration requirement is to select the Planet with the minimum $f_{i,j}$ to access and the optimal data upload amount is $x_{i,j}^*$.

For user groups with task migration requirements ($m_i \neq 0$), their payoff can be expressed as follows,

$$\Pi_i^{ug} = \begin{cases} U_i(x_{i,j}) - f_{i,j}x_{i,j} - d_{j',j}m_i\beta, & \text{if } j \neq j', \\ U_i(x_{i,j}) - f_{i,j}x_{i,j}, & \text{if } j = j', \end{cases} \tag{8}$$

where j' represents the Planet chosen to access by the user group i in the current time period. For a given j, the term $d_{j',j}m_i\beta$ is a constant, so the optimal data upload amount remains the same as analyzed earlier, which is $x_{i,j}^*$. Each user group substitutes $x_{i,j}^*$ into Π_i^{ug}, and selects the Planet corresponding to the maximum payoff as the Planet to be accessed in the next time period. □

After obtaining the optimal Planet access scheme, each group calculates the bid $w_{i,j,r}^s = p_{j,r}c_{s,r}^j r_{i,s}x_{i,j}^*$, which represents the leasing payment of resource r that group i submits to Planet j in order to carry out activity s.

B. Resource Pricing Scheme of Planet

Based on the received bids from all user groups, each Planet updates the prices of all kinds of resources. We present the resource pricing scheme of each Planet in Theorem 2.

Theorem 2. *The optimal resource pricing scheme, which meets the three constraints in Sect. II-E, is shown as follows,*

$$\hat{p}_{j,r} = \max\{q_{j,r}, \frac{\sum\limits_{i,s} A_{i,t,s}\frac{w_{i,j,r}^s}{D_{i,s}}}{C_{j,r}} | t = 1, 2, ..., Z\}. \tag{9}$$

Proof of Theorem 2. By employing such a resource pricing scheme, we can conclude that the resource price $p_{j,r}$ is no less than $q_{j,r}$, satisfying the first constraint. Additionally, the proposed resource pricing scheme does not have any additional conditions that would cause price growth, thus satisfying the third constraint as well. We now focus on how it satisfies the second constraint, which is the capacity constraint.

After updating the resource prices, the actual data amount $\hat{x}_{i,j}$ that can be processed satisfies the following inequality:

$$\hat{x}_{i,j}r_{i,s}c_{s,r}^j\hat{p}_{j,r} \leq x_{i,j}^* r_{i,s}c_{s,r}^j p_{j,r}, \ \forall i, j, r, s. \tag{10}$$

The above inequality can be further processed as follows,

$$\sum_{i,s}\frac{\hat{x}_{i,j}r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}\hat{p}_{j,r} \leq \sum_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}p_{j,r}, \forall j, r, t, \tag{11}$$

$$c_{j,r,t}^{allocated} \leq \frac{\sum\limits_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}p_{j,r}}{\hat{p}_{j,r}}, \forall j, r, t, \tag{12}$$

Algorithm 1: The DDRA Algorithm

Input: Resource prices $\mathbf{p} = \{p_{j,r}|\forall j, r\}$ and Planet access scheme of each group $\mathbf{x}_i = \{x_{i,j}|\forall j\}$ in the current time period, timetable $\mathcal{T}$ of each group for the next time period;

Output: Resource prices $\hat{\mathbf{p}} = \{\hat{p}_{j,r}|\forall j, r\}$ and Planet access scheme of each group $\hat{\mathbf{x}}_i = \{\hat{x}_{i,j}|\forall j\}$ in the next time period;

Initialization: ϵ, $\mathbf{x}_i^*$, $\hat{\mathbf{x}}_i \leftarrow \mathbf{x}_i$
while $\|\hat{\mathbf{x}} - \mathbf{x}^*\| > \epsilon$ **do**
  for *user group* $i = 1$ *to* I **do**
    if i *have no task migration requirement* **then**
      Access the Planet with the minimum $f_{i,j}$;
    else
      Access the Planet with the maximum Π_i^{ug}, according to (8);
    Obtain $x_{i,j}^*$ via (6);
    To prevent oscillations, let

$$x_{i,j} = \gamma x_{i,j}^* + (1-\gamma)\hat{x}_{i,j}, 0 < \gamma < 1; \tag{16}$$

    Make bids: $w_{i,j,r}^s = p_{j,r}c_{s,r}^j r_{i,s}x_{i,j}$;
  for *Planet* $j = 1$ *to* J **do**
    Receive bids from user groups and calculate (9) to obtain the updated resource price $\hat{p}_{j,r}$;
    Broadcast $\hat{p}_{j,r}$ to all groups;
  for *user group* $i = 1$ *to* I **do**
    According to (10), calculate the actual amount of data that can be processed:

$$\hat{x}_{i,j} = \min_{\mathbf{r}} \frac{p_{j,r}}{\hat{p}_{j,r}}x_{i,j}^*; \tag{17}$$

  Calculate $\|\hat{\mathbf{x}} - \mathbf{x}^*\|$.

where $c_{j,r,t}^{allocated}$ represents the amount of resource r allocated by Planet j at time slot t, satisfying

$$c_{j,r,t}^{allocated} = \sum_{i,s}\frac{\hat{x}_{i,j}r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}. \tag{13}$$

According to the resource pricing scheme, we have

$$\hat{p}_{j,r} \geq \frac{\sum\limits_{i,s} A_{i,t^*,s}\frac{w_{i,j,r}^s}{D_{i,s}}}{C_{j,r}} = \frac{\sum\limits_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t^*,s}}{D_{i,s}}p_{j,r}}{C_{j,r}}, \forall j, r, \tag{14}$$

where $t^* = \arg\max \frac{\sum\limits_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}p_{j,r}}{C_{j,r}}$. As such, we have

$$c_{j,r,t}^{allocated} \leq \frac{\sum\limits_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t,s}}{D_{i,s}}p_{j,r}}{\sum\limits_{i,s}\frac{x_{i,j}^* r_{i,s}c_{s,r}^j A_{i,t^*,s}}{D_{i,s}}p_{j,r}}C_{j,r} \leq C_{j,r}, \ \forall j, r, t, \tag{15}$$

which satisfies the resource capacity constraint. □

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

C. Distributed Dynamic Resource Allocation (DDRA) Algorithm

Based on the discussions in Sect. III-A and Sect. III-B, the overall DDRA algorithm is presented in Algorithm 1. The algorithm is executed before the start of each time period. During each execution, it first undergoes initialization, and then proceeds in three phases. In the first phase, each user group determines its optimal Planet access scheme and submits bids to the corresponding Planet. In the second phase, each Planet updates its resource pricing scheme based on the received bids and then broadcasts the prices to all groups. In the third phase, each user group calculates the actual amount of data that can be processed, denoted as $\hat{x}_{i,j}$, according to (10). If $\|\hat{\mathbf{x}} - \mathbf{x}^*\| > \epsilon$, the process returns to phase one and user groups continue to update their strategies.

IV. Validation

In this section, we evaluate the performance of the proposed algorithm using simulations in MATLAB.

A. Simulation Setup

Scenario Description. The simulation scenario comprises 1 central cloud server, 3 Planets, and 200 user groups, with each user group consisting of 2 to 50 members. We consider a 1000×1000 grid and place the central cloud server, Planets, and all users on the grid points. Considering practical scenario, we divide the user groups into two categories: centralized and decentralized. In the centralized setting, 80% of the members are distributed in proximity, while the remaining 20% are randomly distributed. In the decentralized setting, members of the user group are randomly distributed throughout the grid. The Manhattan distance between two grid points represents the network distance between two devices.

User Group Parameter Setting. We generate activity schedules for 200 user groups in time periods $1\sim3$, with each period divided into 10 time slots, where each time slot represents 1 second. It is assumed that the amount of data generated by a user within a time slot follows a Gaussian distribution, and its mean and standard deviation are related to the type of social activity involved. The mean and standard deviation are uniformly distributed within the intervals of $[30, 50]$ Mb and $[0.5, 2]$, respectively. We use the distribution value that corresponds to a cumulative distribution function value of 0.7 as an estimate of the data generation volume. The user groups can engage in 6 types of social Metaverse activities, including VR games, remote co-working, virtual music concerts, etc. Since each user group has different QoE requirements for different types of activities, $n_{i,s}$ is uniformly distributed within the interval of $[1, 10] \times 10^{-4}$¢. For user groups who have task migration requirements, the amount of data that needs to be migrated is set to 0.1 of the total amount of data generated during this activity in the previous time period.

TABLE II
Resource Capacities of 3 Planets

Planets	CPU (cores)	RAM (GBytes)	GPU (SMs)	VRAM (GBytes)
Planet 1	96	256	420	120
Planet 2	128	384	420	120
Planet 3	128	384	336	96

Planet Parameter Setting. Planets have 4 types of resources: CPU, RAM, GPU, and VRAM, which are utilized to provide social Metaverse services to user groups. For 1Mb/s of data traffic, the values of consumption vector $c^j_{s,r}$ are taken according to uniform distributions in the range of $[0.004, 0.008]$ cores for CPU, in the range of $[0.01, 0.02]$ GB for RAM [23], in the range of $[0.02, 0.04]$ SMs for GPU, and in the range of $[0.006, 0.012]$ GB for VRAM. We assume that each Planet specializes in providing a specific type of service, meaning that the $c^j_{s,r}$ of that service is lower compared to other services and is set as the minimum value within the range of the uniform distribution. The operational expenditures are taken according to uniform distributions in the range of $[0.5, 1]$¢ for unit RAM and VRAM, in the range of $[0.5, 1]$¢ for 1 core of CPU [24], and in the range of $[0.2, 0.4]$¢ for 1 SM of GPU. The resource capacities of Planets are presented in Table II.

Auction Setting. We consider two types of auctions: *Auction Type-1*, where user groups do not have task migration requirements and *Auction Type 2*, where user groups have task migration requirements. Auction Type 1 is typically conducted

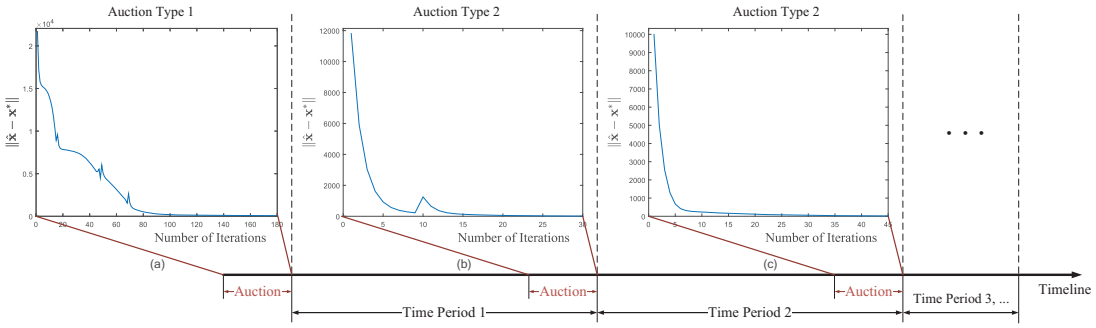

Fig. 3. Convergence of the DDRA algorithm for: (a) Auction Type 1 conducted before time period 1; (b) Auction Type 2 conducted before time period 2; and (c) Auction Type 2 conducted before time period 3.

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

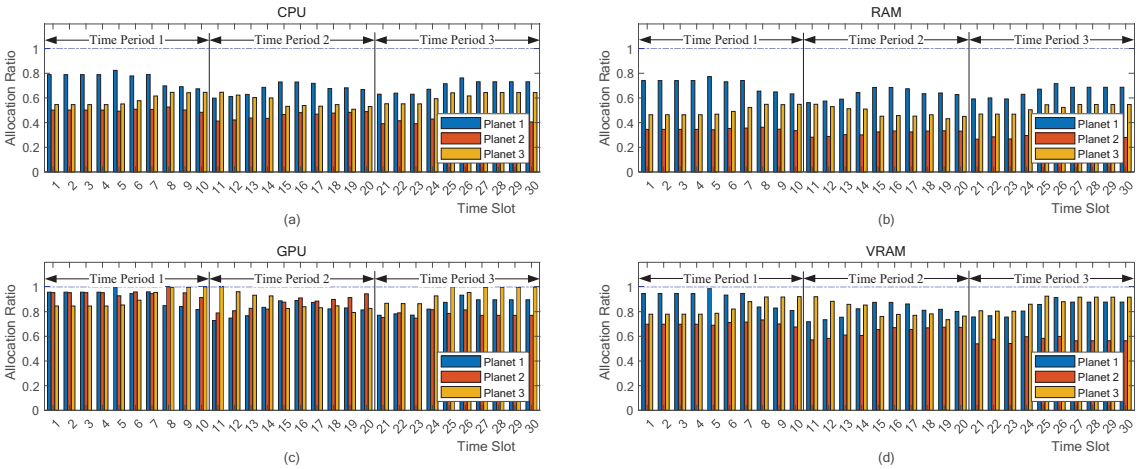

Fig. 4. Resource allocation ratios of (a) CPU; (b) RAM; (c) GPU; and (d) VRAM in Planets 1∼3 over 30 time slots.

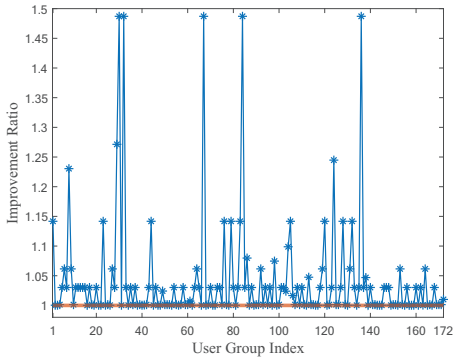

Fig. 5. Improvement ratios of active user groups in time period 1 of the DDRA algorithm compared to the uniform allocation.

before the start of the system, as well as the restart due to factors such as a central cloud server crash or maintenance. Sometimes, during the normal operation of the system, if none of the groups have task migration requirements for the new period, Auction Type 1 will also be conducted. Auction Type 2 is conducted when at least one user group has task migration requirement for the new period. To avoid randomness and validate the generality of the DDRA algorithm, we further consider the dynamics of user group social demands. In the simulation, we not only present the performance results of different auction types but also demonstrate the performance results of the same type of auction (Auction Type 2, which is more common) in different time periods.

B. Numerical Results

Algorithm convergence. In the proposed DDRA mechanism, an auction is conducted before the start of each time period to determine the resource allocation scheme for that time period. Fig. 3 illustrates the convergence of the first three auctions in the system, with $\epsilon = 1$. Auction Type 1 is conducted before time period 1 starts and Auction Type 2 is conducted before time periods 2 and 3 start. In each auction process, as the number of iterations increases, we can observe the gradual reduction in the distance between $\hat{\mathbf{x}}$ and $\mathbf{x}^*$, indicating that the actual Planet access situations of user groups approaches the optimal Planet access schemes. Eventually, user groups no longer update the Planet access schemes, and the algorithm achieves convergence. It is observed that the algorithm converges faster in the auctions conducted before the second and third time periods than that in the auction before the first time period. This is because in the latter two auction processes, some user groups need to consider the problem of task migration, and they are more inclined to not change the access Planet, which helps the algorithm converge. It is worth noting that the proposed algorithm has low computational complexity, and the user group does not need to set aside time for auction, but can synchronize the auction process with the social Metaverse activities.

Allocation effectiveness of multi-dimensional resources. Fig. 4 illustrates the allocation ratios of 4 types of resources among 3 Planets over 30 time slots. The resource allocation ratio is the ratio of the allocated resource to the capacity of this type of resource. The results show that the allocation ratios of the four types of resources change dynamically, which reflects the dynamic social demands of user groups. The difference of resource allocation ratios in different time periods is more obvious, which is mainly due to the different resource pricing schemes and Planet access schemes in different time periods. It is observed that GPU is the fully allocated resource for all three Planets. In each time slot, the allocation ratio of each resource of each Planet is no larger than 1, which indicates that the proposed resource pricing scheme meets the resource capacity constraint and is reasonable and feasible.

The proposed DDRA algorithm outperforms a *uniform allocation* scheme wherein a fraction $\sum_{r,s} w^s_{i,j,r} / \sum_{i,r,s} w^s_{i,j,r}$ of 4 types of resources of Planet j is allocated to user group i at the NE. We evaluate the improvement ratio, which is the ratio between the amount of data each user group can be processed

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.

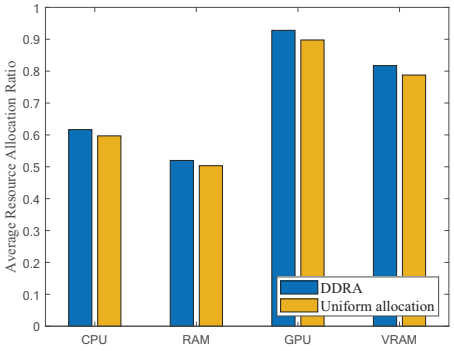

Fig. 6. Resource allocation ratios when averaged over time slot 1∼30 and over all Planets for the DDRA algorithm and the uniform allocation.

by Planet under DDRA algorithm and that under uniform allocation scheme in one time period. In Fig. 5, we show that the improvement ratios of all 172 active user groups (who have social Metaverse activity demand in time period 1) are no less than 1, which ensures a worst-case guarantee for each user group. We further compare the average resource allocation ratios of each resource across time slots 1∼30 and all Planets between DDRA algorithm and the uniform allocation scheme, which are shown in Fig. 6. The results indicate that the DDRA algorithm achieves higher average resource allocation ratios than uniform allocation, leading to a more efficient utilization of resources at Planets.

V. Conclusion

In this paper, we have devised a three-layer network architecture of the Metaverse wherein resource-scarce users can lease resources from Planets to enhance their QoE. To dynamically allocate the various types of limited resources in Planets to user groups, we have proposed the DDRA algorithm, which conducts an auction game among user groups before the start of each time period. Specifically, each user group decides on the optimal Planet access scheme to maximize its payoff and places bids for the needed resources. After receiving bids from user groups, each Planet determines a reasonable resource pricing scheme and then allocates resources to user groups. Simulation results have shown that the proposed algorithm is able to dynamically and efficiently allocate the resources to users for social Metaverse activities, and outperforms the benchmark scheme in terms of improvement ratio for user groups and average resource allocation ratio for Planets.

Acknowledgement

This work was supported in part by NSFC (Nos. 62171352, U22A2029, U20A20175, 62302387, 62101429), and the Fundamental Research Funds for the Central Universities.

References

[1] H. Ning *et al.*, "A survey on the metaverse: The state-of-the-art, technologies, applications, and challenges," *IEEE Internet Things J.*, 2023, early access, doi:10.1109/JIOT.2023.3278329.

[2] Y. Wang, Z. Su, and M. Yan, "Social metaverse: Challenges and solutions," *IEEE Internet Things Mag.*, vol. 6, no. 3, pp. 144–150, 2023.

[3] W. Y. B. Lim *et al.*, "Realizing the metaverse with edge intelligence: A match made in heaven," *IEEE Wireless Commun.*, 2022, early access, doi:10.1109/MWC.018.2100716.

[4] BBC. Fortnite's travis scott virtual concert watched by millions. Accessed: Apr. 24, 2020. [Online]. Available: https://www.bbc.com/news/technology-52410647

[5] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Commun. Surv. Tutor.*, vol. 25, no. 1, pp. 319–352, 2023.

[6] M. Sugimoto, "Extended reality (XR:VR/AR/MR), 3D printing, holography, A.I., radiomics, and online VR tele-medicine for precision surgery," in *Surgery and Operating Room Innovation*. Singapore: Springer, Nov. 2021, pp. 65–70.

[7] T. Huynh-The, Q.-V. Pham, X.-Q. Pham, T. T. Nguyen, Z. Han, and D.-S. Kim, "Artificial intelligence for the metaverse: A survey," *Eng. Appl. Artif. Intell.*, vol. 117, p. 105581, 2023.

[8] Y. Wang, Z. Su, S. Guo, M. Dai, T. H. Luan, and Y. Liu, "A survey on digital twins: Architecture, enabling technologies, security and privacy, and future prospects," *IEEE Internet Things J.*, vol. 10, no. 17, pp. 14 965–14 987, 2023.

[9] Z. Su, Y. Wang, T. H. Luan, N. Zhang, F. Li, T. Chen, and H. Cao, "Secure and efficient federated learning for smart grid with edge-cloud collaboration," *IEEE Trans. Ind. Inf.*, vol. 18, no. 2, pp. 1333–1344, 2022.

[10] Q. Yang, Y. Zhao, H. Huang, Z. Xiong, J. Kang, and Z. Zheng, "Fusing blockchain and AI with metaverse: A survey," *IEEE Open J. Comput. Soc.*, vol. 3, pp. 122–136, 2022.

[11] Y. Wang, Z. Su, J. Ni, N. Zhang, and X. Shen, "Blockchain-empowered space-air-ground integrated networks: Opportunities, challenges, and solutions," *IEEE Commun. Surv. Tutor.*, vol. 24, no. 1, pp. 160–209, 2022.

[12] E. H.-K. Wu, C.-S. Chen, T.-K. Yeh, and S.-C. Yeh, "Interactive medical VR streaming service based on software-defined network: Design and implementation," in *proc. IEEE Int. Conf. Consum. Electron. Taiwan (ICCE-Taiwan)*, Sep. 2020, pp. 1–2.

[13] H. Du *et al.*, "Generative AI-aided optimization for AI-generated content (AIGC) services in edge networks," 2023, *arXiv:2303.13052*.

[14] Y. Wang, Y. Pan, M. Yan, Z. Su, and T. H. Luan, "A survey on ChatGPT: AI-generated contents, challenges, and solutions," *IEEE Open J. Comput. Soc.*, 2023, early access, doi:10.1109/OJCS.2023.3300321.

[15] M. Xu *et al.*, "Sparks of GPTs in edge intelligence for metaverse: Caching and inference for mobile AIGC services," 2023, *arXiv:2304.08782*.

[16] Y. Wang, Z. Su, Q. Xu, R. Li, T. H. Luan, and P. Wang, "A secure and intelligent data sharing scheme for UAV-assisted disaster rescue," *IEEE/ACM Trans. Networking*, 2023, early access, doi:10.1109/TNET.2022.3226458.

[17] F. P. Kelly, A. K. Maulloo, and D. K. H. Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, pp. 237–252, 1998.

[18] Y. Wang, H. Peng, Z. Su, T. H. Luan, A. Benslimane, and Y. Wu, "A platform-free proof of federated learning consensus mechanism for sustainable blockchains," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 12, pp. 3305–3324, 2022.

[19] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 19–32, 2022.

[20] Y. Wang, Z. Su, N. Zhang, and R. Li, "Mobile wireless rechargeable UAV networks: Challenges and solutions," *IEEE Commun. Mag*, vol. 60, no. 3, pp. 33–39, 2022.

[21] Y. Wang, Z. Su, T. H. Luan, J. Li, Q. Xu, and R. Li, "SEAL: A strategy-proof and privacy-preserving UAV computation offloading framework," *IEEE Trans. Inf. Forensic Secur.*, vol. 18, pp. 5213–5228, 2023.

[22] Y. Wang, Z. Su, A. Benslimane, Q. Xu, M. Dai, and R. Li, "Collaborative honeypot defense in UAV networks: A learning-based game approach," *IEEE Trans. Inf. Forensic Secur.*, 2023, early access, doi:10.1109/TIFS.2023.3318942.

[23] H. Halabian, "Distributed resource allocation optimization in 5G virtualized networks," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 3, pp. 627–642, 2019.

[24] J. Khamse-Ashari, G. Senarath, I. Bor-Yaliniz, and H. Yanikomeroglu, "An agile and distributed mechanism for inter-domain network slicing in next generation mobile networks," *IEEE. Trans. Mob. Comput.*, vol. 21, no. 10, pp. 3486–3501, 2022.

Authorized licensed use limited to: Xian Jiaotong University. Downloaded on June 27,2024 at 08:31:06 UTC from IEEE Xplore. Restrictions apply.