The Fisher Market Equilibrium Price Based Multi-Access Edge Computing Coalition Formation

Xiaolong Wang

Lanzhou Jiaotong University <https://orcid.org/0000-0001-6137-7336>

Jianwu Dang (✉ dangjw25300@163.com)

Lanzhou Jiaotong University

Shuxu Zhao

Lanzhou Jiaotong University

Zhanping Zhang

Lanzhou Jiaotong University

Yangping Wang

Lanzhou Jiaotong University

Zhanjun Hao

Northwest Normal University

Research Article

Keywords: Multi-access edge computing, Market equilibrium price, Computing resource block, Coalition formation

Posted Date: October 5th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-924666/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

RESEARCH

The Fisher Market Equilibrium Price based Multi-Access Edge Computing Coalition Formation

Xiaolong Wang¹, Jianwu Dang^{1*}, Shuxu Zhao¹, Zhanping Zhang¹, Yangping Wang¹ and Zhanjun Hao²

*Correspondence:

dangjw25300@163.com

¹School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China

Full list of author information is available at the end of the article

Abstract

The emergence of multi-access edge computing (MEC) aims at extending cloud computing capabilities to the edge of the radio access network. As the large-scale IoT services are rapidly growing, a single edge infrastructure provider (EIP) may not be sufficient to handle the data traffic generated by these services. The coalition method has been used in MEC for resource optimization, latency, energy consumption reduction, computation offloading, etc. However, the majority of research does not consider the price of computing resources corresponded to a container. Moreover, each SP does not choose EIP with the highest cost-performance to sign a medium/long-term computing resource purchase or lease contract. In this work, we consider a scenario with a collection of SPs with different budgets and several EIPs distributed in geographical locations. During the first phase, we get the market equilibrium price and select the optimal EIPs to make a deal by solving the Eisenberg-Gale convex program. In the second stage, using a mathematical model, we maximize EIP's profits and form stable coalitions between EIPs by a distributed coalition formation algorithm. Numerical results demonstrate that the effectiveness of our method is significantly better than the existing model.

Keywords: Multi-access edge computing; Market equilibrium price; Computing resource block; Coalition formation

Introduction

Multi-access edge computing (MEC) has attracted much attention in academia and industry in enabling ultra-low latency requests, high-bandwidth, and nimble mobile

¹services [1, 2, 3]. In MEC, the computing resource is deployed on the edge side,¹
²closing to users. In contrast to the service centralization of mobile cloud computing,²
³the computation requests from users can be offloaded to edge nodes (ENs) closing to³
⁴users. As fusion of wireless communications and mobile computing, MEC is viewed⁴
⁵as a key technology of next generation networks, e.g., 5G, Internet of Things (IoT),⁵
⁶Internet of Me, Tactile Internet, Social Networks, etc. [4]. However, the power supply⁶
⁷limitation of a single edge node raises the question of how an edge node provides⁷
⁸high computing resources. As the large-scale IoT services, such as healthcare [5],⁸
⁹smart cities [6], agriculture monitoring [7], and many others [8] are rapidly growing,⁹
¹⁰a single EN structure may not be sufficient to handle these data traffic. In addition,¹⁰
¹¹cloud data centers (DCs) are often geographically distant from the end-user, which¹¹
¹²may cause high backhaul traffic. It may induce that the latency constraints of¹²
¹³offloaded computation tasks cannot be satisfied due to the propagation delay from¹³
¹⁴the ENs to the DCs. The above reasons drive physical resources and workload¹⁴
¹⁵sharing between ENs in a peer-to-peer manner by forming coalitions [9].¹⁵

¹⁶A coalition is a collection consists of participants in a game to complete a spe-¹⁶
¹⁷cific task, generating with the arrival of new requests and breaking up with the¹⁷
¹⁸completion of request processing [10, 11]. However, affected by the number of edge¹⁸
¹⁹nodes, computing power, optimization targets, and constraint conditions, the coali-¹⁹
²⁰tion formation is more complicated in natural edge scenarios. Hence, it is one of the²⁰
²¹significant challenges in the field of edge computing [12, 13]. We currently consider²¹
²²a situation where service requests are hosted into containers executed through a²²
²³virtualization platform running on ENs. The objective of each service is to single²³
²⁴out the edge infrastructure providers (EIPs) with maximized cost-effectiveness and²⁴
²⁵find its price in line with market laws under the premise of guaranteeing its QoS.²⁵
²⁶Furthermore, the EIPs' goal is to maximize their profit by amortizing their costs,²⁶
²⁷which requires that the aggregated request load submitted to containers must be²⁷
²⁸intense enough to amortize its costs.²⁸

²⁹The objective of services induces a novel market-based solution framework that²⁹
³⁰aims to get the EIPs maximized cost-effectiveness and obtain a price in line with³⁰
³¹the market laws. The basic idea of the method used in this work is to assign dif-³¹
³²ferent prices to a computing resource block (CRB) corresponding to a container of³²
³³different EIPs. According to the market rules, highly in-demand CRBs are priced³³

¹high while under-demanded CRBs' prices are low. Moreover, we assume that each¹
²SP has a specific budget for CRB procurement. And the budget is used to capture²
³service priority. Given the CRBs prices, each SP buys CRBs from the EIPs with the³
⁴maximized cost-effectiveness. When each SP spends the total budgets and CRBs of⁴
⁵EIPs are entirely sell out, the resulting prices form a market equilibrium. ⁵

⁶ Aims at the second goal, if each EIP agrees to cooperate by sharing their requests⁶
⁷and computing resources, they can increase their profits. In particular, an EIP⁷
⁸can increase its net profit by turning off some ENs and transferring its workloads⁸
⁹to other EIPs, or running workloads originated from other EIPs to improve its⁹
¹⁰revenues. Here, depending on the market price and optimal EIPs for SPs, we use¹⁰
¹¹a suitable algorithm to decide what conditions the EIPs willing to cooperate. The¹¹
¹²main contributions of this work are summarized as follows: ¹²

- ¹³ • We consider a scenario with a collection of SPs with different budgets and sev-¹³
¹⁴eral EIPs distributed in geographical locations. For the scenario, we introduce¹⁴
¹⁵a system architecture of MEC in an area. ¹⁵
- ¹⁶ • By solving the Eisenberg-Gale convex program, each SP chooses EIP with¹⁶
¹⁷the highest cost-performance to sign a medium/long-term computing resource¹⁷
¹⁸purchase or lease contract and gets the market equilibrium price of the com-¹⁸
¹⁹puting resource block. ¹⁹
- ²⁰ • Depending on the results of the above phase, we use a mathematical model to²⁰
²¹maximize EIP's profit and a distributed algorithm to form a stable coalition. ²¹
- ²² • We conduct the simulation experiments for the above question and demon-²²
²³strate that the effectiveness of our method is significantly better than the²³
²⁴existing model. ²⁴

²⁵ The rest of paper is organized as follows. We first discuss related work in Section ,²⁵
²⁶and introduce the system model in Section . In Section , we provide a formulation²⁶
²⁷of the market equilibrium price generating and present the detailed description²⁷
²⁸of the cooperative game. We discuss the simulation results that demonstrate the²⁸
²⁹effectiveness of our approach in Section , followed by the conclusion in Section .²⁹

³¹Related work ³¹

³² The coalition method has already been investigated in recent years to maximize the³²
³³profit of EIP. Notably, the methods based on game theory have also been used in³³

¹the field of mobile/multi-access edge computing for resource optimization [14, 15],¹
²latency and energy consumption reduction [16, 17], computation offloading [18, 19],²
³etc. GlebKoshevoy [20] et al. introduce Shapley value, nucleolus, kernel, and bar-³
⁴gaining set in the cooperative game theory. Using time, utility, cost-performance⁴
⁵[21] stability principle or the reasonableness principle [22], it continuously removes⁵
⁶the relatively unstable coalition structure in the space of strategy and finally retains⁶
⁷the approximately optimal coalition structure. T. Ling [23] et al. proposed a dis-⁷
⁸tributed computation offloading algorithm to achieve the Nash equilibrium of the⁸
⁹game, where the algorithm based on the cooperative game is essential to strive for⁹
¹⁰a relative equilibrium state. X. Wang [24] et al. investigate the dynamic bargain-¹⁰
¹¹ing games with externalities and analyzes the effect of externality on the payoffs of¹¹
¹²players, which shows that externality affects the results of the bargaining game, and¹²
¹³coalition structures affect the payoffs of players. Thus, they are suitable for multi-¹³
¹⁴objective scenarios, especially for solving the optimization problem with economical¹⁴
¹⁵cost and time cost as objectives. T. Li [25] et al. formulate the task scheduling prob-¹⁵
¹⁶lem as a distributed multi-device task scheduling game. It is proved that the task¹⁶
¹⁷scheduling game is a potential game, which possesses a property of finite improve-¹⁷
¹⁸ment and always owns a Nash equilibrium. It solved the cooperative work problem¹⁸
¹⁹of the mobile edge computing based on AD-HOC, verified the conditions to avoid¹⁹
²⁰null solutions. Y. Zhang [26] et al. put forward a dynamic optimization model for²⁰
²¹flexible job-shop scheduling based on cooperative game theory, which provides a²¹
²²new real-time scheduling strategy and method. X. Cao [27] et al design the edge²²
²³federation, an integrated edge computing model, to realize the transparent service²³
²⁴provisioning across independent EIPs and the cloud. They characterize the service²⁴
²⁵provisioning process under their edge federation as a linear programming optimiza-²⁵
²⁶tion model and develop a service provisioning algorithm SEE. However, the majority²⁶
²⁷of research does not consider the price of computing resources corresponded to a²⁷
²⁸container. Even though considering the resource price, it is more in task offloading²⁸
²⁹in MEC. Evidently, the two prices of resources are defined from different application²⁹
³⁰perspectives. Furthermore, many approaches do not choose the EIPs, which max-³⁰
³¹imize SPs' profit, as game players before coalition formation. Different from the³¹
³²existing works, we consider the price of CRB and the EIPs with cost-performance.³²

¹System model

²We introduce the system architecture of MEC in an area that belongs to the different²
³regions in which EIPs and services are distributed. As shown in [Figure](#) , there are³
⁴three layers: the user layer, the aggregation layer, and the MEC layer. First of all,⁴
⁵various end-devices with different computation requirements, such as smartphones,⁵
⁶connected vehicles, set-top-boxes, PCs, sensors, that have different computation⁶
⁷requirements, generate amounts of requests that need to be processed at cloud DCs⁷
⁸or MEC platforms in real-time. Next, the aggregation layer, which comprises of⁸
⁹base stations, switches/routers, access points, is the communication bridge between⁹
¹⁰users and MECs. In other words, requests from the user layer arrive at a Point¹⁰
¹¹of Aggregation (PoA) first and then transmit it to a MEC server for processing.¹¹
¹²Indeed, universities, enterprises, hospitals, etc., can also offload their computation¹²
¹³data to MEC through the aggregation layer. Furthermore, service providers like¹³
¹⁴Netflix, YouTube, and Facebook would purchase many CRBs from EIPs, and then¹⁴
¹⁵install their applications and services onto these EIPs' nodes to serve their customers¹⁵
¹⁶better.¹⁶

¹⁷ We contemplate that EIPs are distributed in different geographic locations in¹⁷
¹⁸an area with distinct configurations and limited computing capacities. Each region¹⁸
¹⁹hosts a set of edge nodes to run the services which are in charge of processing the¹⁹
²⁰data generated by user devices located in that area. We assume that each service²⁰
²¹has a definite budget for CRBs procurement and offloads its requests/data to MEC²¹
²²as many as possible. The service budget may be related to its net profits from users.²²
²³However, we do not focus on it to simplify the problem in this research. Through the²³
²⁴maximum revenue generated by using the resource bundle of EIPs, we can evaluate²⁴
²⁵the merit of an EIP to a service provider. An EIP may have different worth for²⁵
²⁶different service providers because of the varied distance between an SP and an²⁶
²⁷EIP. Intuitively, each SP would choose its favorite resource bundle with optimal²⁷
²⁸cost performance.²⁸

²⁹ There exists a platform between SPs and EIPs for the sake of collecting the²⁹
³⁰information of EIPs (e.g., computing capacities) and SPs (e.g., budget, etc.) and³⁰
³¹computing an ME price of unit resource bundle, which maximize the SPs' revenue³¹
³²and also fully allocates the EIPs' resources. The ME price can act as an incentive³²
³³mechanism to charge SPs and to reward MECs. On one side, if the amount of³³

¹EIPs' resource bundles is fixed, the market-clearing price is increases with the SPs'¹
²budget increase. On the other side, anchored the SPs' budget, the price is inversely²
³proportional to the number of EIPs' resources. To format a rational coalition among³
⁴EIPs, we focus on computing a market-clearing equilibrium price that SPs spent all⁴
⁵budgets and EIPs sold out all resources. 5

6

6

⁷**Problem formulation**7

⁸APs have the same objective as EIPs, which is to prompt their net profit as much⁸
⁹as possible. Without loss of generality, we assume that it is independent among⁹
¹⁰areas to allocate containers/CRBs for services. We only consider the single geo-¹⁰
¹¹graphic district, and extend it to multiple zones is straightforward. In general, we¹¹
¹²deem the EIPs' total computing capacity is steady, which is determined by pre-¹²
¹³dicting the whole region's resource requirements. That is not the point on which¹³
¹⁴we focus. Nevertheless, the requests from users are dynamic variance along with¹⁴
¹⁵the users' changing. Therefore, precisely predicting the demands originated from¹⁵
¹⁶users is challenging. Moreover, to improve Qos and decrease EIPs' expenditure as¹⁶
¹⁷much as possible, the MEC coalition is indispensable. Namely, sharing workloads¹⁷
¹⁸and resource bundles to serve SPs can reduce EIPs' energy consumption costs and¹⁸
¹⁹increase their revenues by hosting services on others' EIP. It is noteworthy that¹⁹
²⁰each EIP prefers to cooperate with some EIP to gain more profit than others. The²⁰
²¹coalition formed is unstable without the above incentive mechanism. That is to say,²¹
²²a member in an alliance can always find a more profitable confederation that leaves²²
²³the current league. 23

²⁴**EIP Selection and CRB pricing model**24

²⁵We denote $\mathcal{M} = \{1, 2, \dots, M\}$ and $\mathcal{S} = \{1, 2, \dots, S\}$ as the set of EIPs and SPs,²⁵
²⁶where $|\mathcal{M}| = M$, $|\mathcal{S}| = S$, respectively. Denote i and j as the SP and EIP index,²⁶
²⁷respectively. Assuming that each EIP j has c_j homogeneous computing unit [28]²⁷
²⁸(e.g., VMs, servers, etc.). Let a_{ij} is the portion of the resources on EIP j allocated²⁸
²⁹to SP i , $0 \leq a_{ij} \leq 1$. Then, $c_j a_{ij}$ is the number of computing units allocated to SP²⁹
³⁰ i from EIP j . Denote $\mathbf{a}_i = (a_{i,1}, a_{i,1}, \dots, a_{i,M})$ as the vector of resources allocated³⁰
³¹to SP i from all the EIPs. We assume that the computing resource is dividable to³¹
³²support hosting container (VM/Docker). Therefore, $c_j a_{ij}$ can be non-integer; and³²
³³we define the budget of SP i as B_i . 33

¹ The ultimate goal is choosing an EIP with the best cost performance for an SP.¹
²Computing ME solutions include an equilibrium price vector $\mathbf{p} = (p_1, p_2, \dots, p_M)$,²
³ p_j is the price of EIP j , and a resource allocation matrix $\mathcal{A}$, the element in which³
⁴at row i th and column j th is a_{ij} . Note that p_j is the price of all the resources⁴
⁵of EIP j . Then one unit resource price equals $\frac{p_j}{c_j}, \forall j$. We define $u_i(\mathbf{a}_i, \mathbf{p})$ as the⁵
⁶SP i 's utility function of the number of resources a_i received from EIPs under⁶
⁷the resource price $\mathbf{p}$. Depend on the capacity constraints of EIPs, then we have⁷
⁸ $\sum_{i=1}^S c_j a_{ij} \leq c_j, \forall j \in \mathcal{M}$ and $\sum_{i=1}^S a_{ij} \leq 1, \forall j \in \mathcal{M}$. Each SP is a player in the⁸
⁹Fisher market game, in which the player aims to maximize its utility subject to the⁹
¹⁰SP budget constraint $\sum_j a_{ij} p_j \leq B_i$.¹⁰

¹¹
¹²**Definition 1** A market outcome that maximizes the utility of each buyer subject¹²
¹³to its budget constraint and clears the market is called a market equilibrium [29].¹³

¹⁴
¹⁵In-state, $(\mathbf{p}^*, \mathbf{A}^*)$ is a market equilibrium if and only if satisfy the following con-¹⁵
¹⁶ditions:¹⁶

- ¹⁷• *Condition 1*: For all $i \in \mathcal{S}$, $\mathbf{a}_i$ maximizes buyers i 's utility given prices $\mathbf{p}^*$ and¹⁷
¹⁸budget B_i , i.e., $\mathbf{a}_i^* = (a_{i,1}^*, \dots, a_{i,M}^*) \in \arg \max_{\sum_j p_j^* a_{ij} \leq B_i; \mathbf{a}_i \geq 0} u_i(\mathbf{a}_i, \mathbf{p}^*)$.¹⁸
- ¹⁹• *Condition 2*: Each item (e.g., computing resource block) j either is completely¹⁹
²⁰sold or has price 0, i.e., $(\sum_{i=1}^S a_{ij} - 1) p_j = 0, \forall j \in \mathcal{M}$.²⁰
- ²¹• *Condition 3*: All budgets get spent, i.e., $\sum_{j=1}^M p_j a_{ij} = B_i, \forall i \in \mathcal{S}$.²¹

²²A market equilibrium is guaranteed to exist if each item is desired by at least one²²
²³buyer and each buyer desires at least one item [30]. Here, condition one guarantees²³
²⁴that the equilibrium allocation $\mathbf{a}_i^*$ maximizes the utility of SP i at the equilibrium²⁴
²⁵prices $\mathbf{p}^*$ and SPs' budgets constraints B_i . Condition two maximizes the EIPs'²⁵
²⁶resource utilization, i.e., the resources of EIPs are entirely sold in the market. Con-²⁶
²⁷dition three represents that each SP runs out of its budget to purchase resources²⁷
²⁸from EIPs. Moreover, condition two and three is called the market clearing condition²⁸
²⁹[31].²⁹

³⁰*Service provider utility*³⁰

³¹For the sake of representation simplicity, we use $u_i(\mathbf{a}_i)$ to denote the utility of SP³¹
³² i , i.e., $u_i(\mathbf{a}_i, \mathbf{p}) = u_i(\mathbf{a}_i)$. The model considers linear functions for ease of investi-³²
³³gation. We view the delay-sensitive applications and assume that the transmission³³

¹bandwidth is large enough and the data size need transmission is relatively small.¹
²Hence, we can neglect the transmission delay and only consider propagation latency²
³and processing delay [32, 33]. The sum total latency of a request from users con-³
⁴sists of three parts: the round-trip latency between a user and a PoA, such as base⁴
⁵station (BS) or access point (AP), the return journey delay between the PoA and⁵
⁶an EIP's node which hosting the service application, and the processing latency of⁶
⁷request. A detailed depiction is shown in Figure . Note that we only investigate the⁷
⁸model from the aggregation layer to the MEC layer. For the sake of simplicity, we⁸
⁹assume that each service is located at one PoA in a region (If a service's requests⁹
¹⁰from several PoAs, we can take sum over all the PoAs' requests). When there is¹⁰
¹¹more than one PoAs for a service, we take the sum over all PoAs requests processed¹¹
¹²by the MEC. Let T_i^{max} as the maximum latency tolerance of service i , then¹²

$$l_{ij}^n + l_{ij}^p \leq T_i^{max}, \quad \forall i, j. \quad (1)$$

¹⁷Denote λ_{ij}^{max} as the maximum number of requests that EIP j 's nodes can process.¹⁷
¹⁸Evidently, if $l_{ij}^n \geq T_i^{max}$, $\lambda_{ij}^{max} = 0$.¹⁸

²⁰We use M/M/1 queues to model the processing latency and assume the workload²⁰
²¹is assigned among computing units evenly [32, 33, 34, 35]. The processing latency²¹
²² l_{ij}^p can be represented as²²

$$l_{ij}^p = \frac{1}{\mu_{ij} - \frac{\lambda_{ij}}{c_j a_{ij}}}, \quad \forall i, j \quad (2)$$

²⁷, where μ_{ij} is the processing rate of a computing unit of EIP j to deal with service²⁷
²⁸ i 's requests, λ_{ij} is the arrival rate of request from service i to EIP j . In addition, to²⁸
²⁹assure the stability of the queue, the inequality $\frac{\lambda_{ij}}{a_{ij}} < \mu_{ij}$ must hold. Otherwise, the²⁹
³⁰delay of the queue will be infinite. Combining formulas (1) and (2), we can derive³⁰

$$\lambda_{ij} \leq c_j a_{ij} \left(\mu_{ij} - \frac{1}{T_i^{max} - l_{ij}^n} \right), \quad \forall i, j. \quad (3)$$

¹If $l_{ij}^n < T_i^{max}$, the maximum number of requests from service i processed by the

²EIP j is

$$\begin{aligned} \lambda_{ij}^{max} &= \max \left\{ c_j a_{ij} \left(\mu_{ij} - \frac{1}{T_i^{max} - l_{ij}^n} \right), 0 \right\} \\ &= c_j a_{ij} \cdot \underbrace{\max \left\{ \left(\mu_{ij} - \frac{1}{T_i^{max} - l_{ij}^n} \right), 0 \right\}}_{g_{ij}}, \forall i, j. \end{aligned} \quad (4)$$

Denote r_i as the profit of successfully serving a request from service i [34]. Hence, the revenue of service i $u_{ij}(a_{ij}) = r_i g_{ij} c_j a_{ij}$. Let $v_{ij} = r_i g_{ij} c_j$, and then we have

$$u_{ij}(a_{ij}) = v_{ij} a_{ij}, \quad \forall i, j. \quad (5)$$

¹³Thus,

$$u_i(\mathbf{a}_i) = \sum_{j=1}^M u_{ij}(a_{ij}) = \sum_{j=1}^M v_{ij} a_{ij}, \quad \forall i. \quad (6)$$

¹⁷Obviously, $r_i g_{ij}$ can be a valuation index which is the gain of service i from one ¹⁷unit resource of EIP j . Accordingly, v_{ij} is a valuation index of all resources of EIP ¹⁸ j .

²¹**Definition 2** A function $u(\cdot)$ is homogeneous of degree d , where d is a constant, ²¹if $u(\alpha x) = \alpha^d u(x)$, $\forall \alpha > 0$ [36].

²⁴It is simple to verify that $u_i(\mathbf{a}_i)$ is a linear function with degree $d = 1$.

²⁷*Solution*

²⁸Define the ratio $\frac{v_{ij}}{p_j}$ is the bang per buck [37] of EIP j to SP i , which is the utility ²⁸gained by SP i per unit amount of money spent on EIP j . Thus, we can define ²⁹ $\alpha(i) = \underset{j}{\operatorname{argmax}} \{ \frac{v_{ij}}{p_j} \}$ as the maximum bang per buck (MBB) [37]. Only SP i buys ³⁰resources from EIP of $\alpha(i)$, will each SP spend entire budget and maximize its ³¹utility. Derive from [definition 2](#), the buyers' utility is linear. Hence, by solving the ³²Eisenberg-Gale (EG) convex program, we can find the market equilibrium. The EG ³³

¹convex program can be described as

$$\begin{aligned}
 & \text{maximize} \sum_{i=1}^S B_i \ln u_i & (7) \\
 \text{s.t.} \quad & u_i = \sum_{j=1}^M v_{ij} a_{ij}, \quad \forall i & (8) \\
 & \sum_{i=1}^S a_{ij} \leq 1, \quad \forall j & (9) \\
 & a_{ij} \geq 0, \quad \forall i, j. & (10)
 \end{aligned}$$

By setting $a_{ij} > 0$ and its value is small enough such that the constraints (8)-(10) are stringent inequality. Thus, Slater's condition holds, and then Karush-Kuhn-Tucker (KKT) conditions are necessary and adequate for optimality [38]. Using φ_i , p_j , and ζ_{ij} to represent the dual variables affiliated with restrictions (8), (9), and (10), severally. According to the Lagrangian, we have

$$\begin{aligned}
 L(u, A, \varphi, p, \zeta) = & \sum_i B_i \ln u_i + \sum_j p_j \left(1 - \sum_i a_{ij} \right) + \\
 & \sum_i \varphi_i \left(\sum_j v_{ij} a_{ij} - u_i \right) + \sum_i \sum_j \zeta_{ij} a_{ij}
 \end{aligned} \tag{11}$$

²²The KKT conditions are

$$\frac{\partial L}{\partial u_i} = 0 \Rightarrow \frac{B_i}{u_i} = \varphi_i, \quad \forall i \tag{12}$$

$$\frac{\partial L}{\partial a_{ij}} = 0 \Rightarrow p_j - B_i \frac{v_{ij}}{u_i} = \zeta_{ij}, \quad \forall i, j \tag{13}$$

$$u_i = \sum_j v_{ij} a_{ij}, \quad \forall i \tag{14}$$

$$p_j \left(1 - \sum_i a_{ij} \right) = 0, \quad \forall j \tag{15}$$

$$\zeta_{ij} a_{ij} = 0, \quad \forall i, j \tag{16}$$

$$p_j \geq 0, \quad \forall j \tag{17}$$

$$\zeta_{ij} \geq 0, \quad \forall i, j. \tag{18}$$

¹From the KKT conditions, we can derive

$$\frac{u_i}{B_i} \geq \frac{v_{ij}}{p_j}, \quad \forall i, j \quad (19)$$

$$if \quad a_{ij} > 0 \Rightarrow \zeta_{ij} = 0 \Rightarrow \frac{u_i}{B_i} = \frac{v_{ij}}{p_j}, \quad \forall i, j \quad (20)$$

$$p_j > 0 \Rightarrow \sum_i a_{ij} = 1, \quad \forall j \quad (21)$$

$$\sum_i a_{ij} < 1 \Rightarrow p_j = 0, \quad \forall j \quad (22)$$

⁹Note that variable p_j can be seen in the EG convex program as the pang per buck⁹
¹⁰of EIP j to SP i . The subsequent theorem reveals the connection between the EG¹⁰
¹¹convex program and the market equilibrium solution. Besides, some properties of¹¹
¹²the ME are also represented.

¹⁴**Theorem 1** *The optimal solution of the EG convex program is a market equilib-*
¹⁵*rium. Particularly, the Lagrangian dual variables of the EIPs' capacity restrictions*
¹⁶*(9) are the equilibrium prices. In addition, the allocation at the equilibrium is Pareto*
¹⁷*optimal and envy free, which also satisfies the sharing-incentive and proportionality*
¹⁸*properties [39].*

²⁰At the equilibrium, the resource allocation maximizes the utility and exhausts²⁰
²¹each service's budget. Moreover, each SP i only purchases resources from ENs in²¹
²² α_i . Additionally, SPs' optimal utilities and equilibrium prices are exclusive.

²⁴Optimal Coalition Allocation Utility

²⁵An EIP j aims at increasing its net margin as much as possible. We assume that²⁵
²⁶each service application is encapsulated in a container with a computing resource²⁶
²⁷block, and each EIP's edge nodes need a virtualization platform to run the container.²⁷
²⁸ $EN(j)$ represents the set of edge nodes of EIP j . Let $w_z(\eta)$ be the expected power²⁸
²⁹consumption of edge node z .

$$w_z(\eta_z) = \eta_z W_z^{max} + (1 - \eta_z) W_z^{min} \quad (23)$$

³², where $\eta \in [0, 1]$ is the utilization of resource (CPU) on EIP edge node z , W_z^{max} ³²
³³and W_z^{min} represent the power consumption with idle and fully utilized CPU states,³³

respectively. The net margin rate of EIP $\alpha(i)$ denoted by $M_{\alpha(i)}$.

$$\begin{aligned}
 M_{\alpha(i)} &= R_{\alpha(i),i} n_i - \left(P_{i,\alpha(i)} b_{i(n_i < N_i)} + \sum_{e \in EN(\alpha(i))} x_e w_e(\eta_e) E_{\alpha(i)} \right) \\
 &= R_{\alpha(i),i} n_i - \left(P_{i,\alpha(i)} b_{i(n_i < N_i)} + \sum_{e \in EN(\alpha(i))} W_e(x_e, \eta_e) E_{\alpha(i)} \right)
 \end{aligned} \tag{24}$$

, where $W_e(x_e, \eta_e) = x_e W_e^{min} + (W_e^{max} - W_e^{min}) \eta_e$; x_e is a binary variable that depicts the edge node state; when $x_e = 1$, the state is switched on; conversely $x_e = 0$, the state is switched off. n_i denotes the number of containers/CRBs allocated for service i , and N_i is the minimum number of containers/CRBs that satisfies the delay requirement of service i , which can derive from the inequality (4) and the stability condition $\frac{\lambda_{ij}}{a_{ij}} < \mu_{ij}$ of the queue. η_e represents the ratio of resources allocated to all the containers running on edge node e to its total resource capacity. E_j is the price (per unit of time) charged EIP j for electricity. b_i is binary variable. When $n_i < N_i$, $b_i = 1$, which represents that CRBs allocated for service i cannot satisfy the Qos requirement, conversely, $b_i = 0$. When SP i 's Qos is not met, it will charge EIP j for a monetary penalty $P_{i,j}$. $R_{\alpha(i),i}$ represents the revenue that EIP $\alpha(i)$ earns for running a container for processing its requests from SP i . In equation (24), the first term on the right means the sum of EIP $\alpha(i)$'s income to run n_i containers, where the second term on the right denotes the overheads of EIP $\alpha(i)$.

The coalition formation algorithm (in Subsection) needs to compute the coalition net margin $\Psi(\mathcal{C})$ to get the payoffs that EIPs gain in any coalition $\mathcal{C}$. By introducing the Mixed Integer Linear Program (MILP) model to solve a maximization problem, we can find the best allocation of the containers $\mathcal{V}$ onto the edge nodes $\mathcal{N}_c = \bigcup_{j \in \mathcal{C}} EN(j)$ to run the services $\mathcal{S}_c = \{i : \alpha(i) \in \mathcal{C}\}$. Denote $m(i)$ as a mapping $\mathcal{V} \rightarrow \mathcal{S}_c$, which represents the service application run by container i . For the sake of simplicity, we define vector $\mathbf{R} = (R_{\alpha(i),i})_{i \in \mathcal{S}_c}$, $\mathbf{n} = (n_i)_{i \in \mathcal{S}_c}$, $\mathbf{b} = (b_i)_{i \in \mathcal{S}_c}$, $\mathbf{P} = (P_{i,\alpha(i)})_{i \in \mathcal{S}_c}$, $\mathbf{W} = (W_j(x_j, \eta_j))_{j \in \mathcal{N}_c}$, $\mathbf{E} = (E_{\alpha(i)})_{\alpha(i) \in \mathcal{C}, j \in EN(\alpha(i))}$ (here, j is the

edge node). The MILP model can be described as

$$\text{maximize } \Psi(\mathcal{C}) = \mathbf{R} \cdot \mathbf{n} - (\mathbf{b} \cdot \mathbf{P} + \mathbf{W} \cdot \mathbf{E}) \quad (25)$$

$$\text{s.t. } \sum_{i \in \mathcal{N}_c} y_{ij} \leq 1, \quad \forall j \in \mathcal{V} \quad (26)$$

$$\sum_{j \in \mathcal{V}} y_{ij} \leq |\mathcal{V}| x_i, \quad \forall i \in \mathcal{N}_c \quad (27)$$

$$\eta_i = \sum_{j \in \mathcal{V}} y_{ij} \frac{1}{L_i}, \quad \forall i \in \mathcal{N}_c \quad (28)$$

$$\eta_i \leq x_i, \quad \forall i \in \mathcal{N}_c \quad (29)$$

$$n_k = \sum_{i \in \mathcal{N}_c} \sum_{j \in \mathcal{V}, m(j)=k} y_{ij}, \quad \forall k \in \mathcal{S}_c \quad (30)$$

$$n_k \leq N_k, \quad \forall k \in \mathcal{S} \quad (31)$$

$$x_i \in \{0, 1\}, \quad \forall i \in \mathcal{N}_c \quad (32)$$

$$y_{ij} \in \{0, 1\}, \quad \forall i \in \mathcal{N}_c, j \in \mathcal{V} \quad (33)$$

$$\eta_i \in \mathbb{R}^+, \quad \forall i \in \mathcal{N}_c \quad (34)$$

$$n_k \in \mathbb{N}, \quad \forall k \in \mathcal{S}_c \quad (35)$$

In the above MILP model, the binary variable $x_i = 1$ if edge node i is turned on, otherwise $x_i = 0$. The binary variable $y_{ij} = 1$ if container j is allocated on edge node i , else $y_{ij} = 0$. The non-negative real variable η_i denotes the ratio of resources allocated to all the containers running on edge node i to its total resource capacity, and L_i represents the maximum number of containers that can host on edge node i . The non-negative integer variable n_k denotes the number of containers allocated for service k . The objective function $\Psi(\mathcal{C})$ of the maximization problem is the retained profits rate gained by coalition $\mathcal{C}$ composed of EIPs. It is an extension of the net profit $M_{\alpha(i)}$ of EIP $\alpha(i)$ and has the following constraints.

Constraint condition (26) ensures that each container is hosted on one edge node at most. Limitation condition (27) assures that containers will not be allocated on edge nodes turned off. Equation (28) defines the value of variable η_i as the ratio of the sum of containers distributed to the maximum number of containers that can host on edge node i . Moreover, inequation (29) guarantees that the ratio η_i does not exceed one (edge node i switched on) or zero (edge node i turned off). Equation (30) defines the value of the variable n_i as the number of containers allocated for

¹service k . Restriction condition (31) ensures that the number of containers assigned¹
²to each service cannot exceed the actual demand. Equations (32)-(35) define the²
³domain of variables x_i , y_{ij} , η_i and n_k , respectively. 3

⁴4
⁵Coalition preference function 5

⁶We assume that each EIP is a rational participant in a cooperative game to make⁶
⁷optimal strategic decisions to increase their profits. Thus, every EIP may form a⁷
⁸coalition to pursue its maximum yields, in which each EIP (coalition member) must⁸
⁹go along with sharing its resources with others. An alliance formation is a dynamic⁹
¹⁰process, where an EIP moves from one coalition with a lower utility to another with¹⁰
¹¹a higher. We model the process as a coalition formation cooperative game. Foremost,¹¹
¹²an EIP must ensure that joining a league gained is no worse than working alone.¹²
¹³In addition, to ensure that the benefits are not short-lived, the coalition, an EIP¹³
¹⁴joined, needs to satisfy the following properties [40, 41]. 14

- ¹⁵ • *Coalition stability property*: A coalition is stable if no participant can deviate¹⁵
¹⁶from the current federation to reach a better one subjectively. 16
- ¹⁷ • *Fairness property*: Any member in the current coalition expects that the re-¹⁷
¹⁸sulting profits divide fairly among all players. 18

¹⁹Hence, the EIPs desiderate finding a method to decide whether to join a coalition.¹⁹
²⁰By modeling the coalition formation process as a cooperative game with transferable²⁰
²¹utility [42], each EIP can cooperate with others to maximize its net profit by using a²¹
²²coalition formation algorithm. Here, we use a hedonic game, where any player's gain²²
²³is decided by the coalition members to which the player belongs; and the players'²³
²⁴preferences for their possible set of coalitions form the final alliances. In other terms,²⁴
²⁵each participant does not consider other coalition players how grouped, and is only²⁵
²⁶interested in the players which in the coalition he joined. 26

²⁷Given that $\mathcal{M}_s$ is the EIPs set selected, then a coalition $\mathcal{C} \subseteq \mathcal{M}_s$ acts as an²⁷
²⁸entity, representing an agreement that they must agree to share their resources²⁸
²⁹and users among EIPs. At any time, we divide the participants set into a coalition²⁹
³⁰partition $\Pi = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_l\}$. Each element $\mathcal{C}_k$ in partition Π is disjoint for $k =$ ³⁰
³¹ $1, 2, \dots, l$, i.e., $\cup_{k=1}^l \mathcal{C}_k = \mathcal{M}_s$ and $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$, $i \neq j$. Denote $\mathcal{C}_\Pi(i)$ as a coalition³¹
³²containing player i in partition Π , $\forall i \in \mathcal{M}_s$. Define $\Psi(\mathcal{C} | \Pi)$ as the utility value of³²
³³a coalition $\mathcal{C}$ in partition Π of $\mathcal{M}_s$. For a hedonic game, the utility value of alliances³³

¹is independent of each other, i.e., $\Psi(\mathcal{C} \mid \Pi) = \Psi(\mathcal{C})$, which can be calculated by¹
²formula (25) in subsection . Let $\phi_i(\mathcal{C}, \Psi)$ be a fraction coalition utility value received²
³from coalition $\mathcal{C}$ by i , where $i \in \mathcal{C}$. That is to say, $\phi_i(\mathcal{C}, \Psi)$ is the payoff of i in³
⁴coalition $\mathcal{C}$. Each player in cooperative games aims to get a stable coalition and⁴
⁵gains rewards as high as possible. In parallel, it is critical to maintaining a stable⁵
⁶coalition to distribute the federation value fairly among the members. Furthermore,⁶
⁷every participant cannot improve its rewards by leaving the current coalition to join⁷
⁸a new one. In a nutshell, the coalition value distribution is fair and can be reached⁸
⁹using a profit allocation schedule with the following properties [43].⁹

- ¹⁰ • *Efficiency property*: $\sum_{i \in \mathcal{C}} \phi_i(\mathcal{C}, \Psi) = \Psi(\mathcal{C})$.¹⁰
- ¹¹ • *Symmetry property*: If $\forall i, j \in \mathcal{M}_s, \forall \mathcal{C} \subseteq \mathcal{M}_s \setminus \{i, j\}$ and $\Psi(\mathcal{C} \cup \{i\}) =$ ¹¹
¹² $\Psi(\mathcal{C} \cup \{j\})$, then $\phi_i(\mathcal{C} \cup \{i\}, \Psi) = \phi_j(\mathcal{C} \cup \{j\}, \Psi)$.¹²
- ¹³ • *Dummy player*: If i is a dummy EIP, i.e., $\Psi(\mathcal{C} \cup \{i\}) - \Psi(\mathcal{C}) = 0$ for $\forall \mathcal{C} \subseteq$ ¹³
¹⁴ $\mathcal{M}_s \setminus \{i\}$, then $\phi_i(\mathcal{C}, \Psi) = 0$.¹⁴
- ¹⁵ • *Fairness*: If $\forall i, j \in \mathcal{M}_s, \forall \mathcal{C} \subseteq \mathcal{M}_s \setminus \{i, j\}$ and $\Psi(\mathcal{C} \cup \{i\}) = \Psi(\mathcal{C} \cup \{j\})$, then¹⁵
¹⁶ $\phi_i(\mathcal{C} \cup \{i, j\}, \Psi) - \phi_i(\mathcal{C} \cup \{i\}, \Psi) = \phi_j(\mathcal{C} \cup \{i, j\}, \Psi) - \phi_j(\mathcal{C} \cup \{j\}, \Psi)$.¹⁶

¹⁷ The efficiency property guarantees that the total coalition value is assigned. The¹⁷
¹⁸ symmetry property denotes that if two participants who make the same contribute¹⁸
¹⁹ to every subset composed by other players, they should obtain the same profit.¹⁹
²⁰ The dummy property represents that the members, nothing marginal contribution²⁰
²¹ provided to any other coalitions, get zero profit. In addition, the fairness property²¹
²² means that any two participants who make the same contribution to the alliance will²²
²³ get the same payoff. Shapely value [44], a payoff allocation, satisfies the additivity,²³
²⁴ strong monotonicity and the above four properties. Here, we use the Shapely value²⁴
²⁵ defined in the literature [45], described as²⁵

$$\begin{aligned}
 \phi_i(\mathcal{C}, \Psi) &= \sum_{\mathcal{B} \subseteq \mathcal{C} \setminus \{i\}} \frac{|\mathcal{B}|! (|\mathcal{C}| - |\mathcal{B}| - 1)!}{|\mathcal{C}|!} (\Psi(\mathcal{B} \cup \{i\}) - \Psi(\mathcal{B})) \\
 &= \sum_{\mathcal{B} \subseteq \mathcal{C} \setminus \{i\}} \left[\binom{|\mathcal{C}|}{|\mathcal{B}|} (|\mathcal{C}| - |\mathcal{B}|) \right]^{-1} (\Psi(\mathcal{B} \cup \{i\}) - \Psi(\mathcal{B}))
 \end{aligned}
 \tag{36}$$

³², where $\mathcal{B}$ is all possible subsets that do not contain i of coalition $\mathcal{C}$, including the³²
³³ empty set.³³

¹ Define $\succeq_i$ as a preference relation that EIP i uses to compare all the possible¹
² coalition that it may join. The binary relationship needs to satisfy complete, reflex-²
³ ive, and transitive, etc., properties [41]. In other terms, $\forall \mathcal{C}_1, \mathcal{C}_2 \subseteq \mathcal{M}_s$, where $i \in \mathcal{C}_1$ ³
⁴ and $i \in \mathcal{C}_2$, $\mathcal{C}_1 \succeq_i \mathcal{C}_2$ represents that participant i prefers being a member of $\mathcal{C}_1$ ⁴
⁵ than $\mathcal{C}_2$, or at least i prefers both equally. Hence, we use the following preference⁵
⁶ relation⁶

$$\mathcal{C}_1 \succeq_i \mathcal{C}_2 \Leftrightarrow \Phi_i(\mathcal{C}_1) \geq \Phi_i(\mathcal{C}_2) \quad (37)$$

⁹ where $\mathcal{C}_1$ and $\mathcal{C}_2$ are any two coalitions that contain member i . $\Phi_i(\cdot)$ is the pref-¹⁰
¹¹ erence function, described as¹¹

$$\Phi_i(\mathcal{C}, \Psi) = \begin{cases} \phi_i(\mathcal{C}, \Psi), & \text{if } \mathcal{C} \notin h(i) \\ -\infty, & \text{otherwise} \end{cases} \quad (38)$$

¹⁴ , where $h(i)$ is a set that contains the coalitions having been evaluated [46, 47].¹⁵
¹⁶ For member i , if $\mathcal{C}_1$ is strictly better than $\mathcal{C}_2$, we use notation $\succ_i$ to denote the¹⁶
¹⁷ preference relationship, i.e., $\mathcal{C}_1 \succ_i \mathcal{C}_2$.¹⁷

¹⁸ Coalition Formation Algorithm¹⁸

¹⁹ After deriving the price of the computing resource block, coalition utility, and pref-¹⁹
²⁰ erence function from subsection -, in this subsection, we use a distributed algorithm²⁰
²¹ named DCFA (Distributed Coalition Formation Algorithm) based on the hedonic²¹
²² shift rule [48] to get a stable coalition. By introducing the suitable distributed state²²
²³ management algorithms [49, 50], coping with the distribution property of DCFA.²³

²⁴ Describing simply the shift rule as, given a coalition partition $\Pi = \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_l\}$ ²⁴
²⁵ on set $\mathcal{M}_s$ and a preference $\succ_i$, if and only if $\mathcal{C}_k \cup \{i\} \succ_i \mathcal{C}_\Pi(i)$, player i leaves its²⁵
²⁶ current coalition $\mathcal{C}_\Pi(i)$ to join coalition $\mathcal{C}_k \in \Pi \cup \emptyset$. The shift rule exploits the²⁶
²⁷ rational participants' selfishness to move from low profit to higher, ignoring the²⁷
²⁸ influence of this behavior on other individuals in the same coalition. The objective²⁸
²⁹ of DCFA is to let each member find the possible alliances it may join and check²⁹
³⁰ whether it is preferred to join. If a member decides to leave the current coalition³⁰
³¹ to join a new one, we put the current coalition into set $h(i)$ to avoid repeatedly³¹
³² visiting it. Given that each EIP can operate asynchronously and independently³²
³³ from others, we can implement DCFA using suitable distributed mechanisms, which³³

¹are state retrieval and atomic state update. The previous one guarantee that any¹
²member can obtain the current coalition partition; the second one updates the²
³coalition partition when any member decides. A similar issue instance can refer to³
⁴the literature [49, 50, 51].⁴

Algorithm 1 DCFA⁵

```

61: function DCF( $eip_i, partition$ )6
72:   Initialize  $h(eip_i)$ ,  $c\_current$ ,  $c\_best$  as an empty set, respectively.7
83:   while true do8
94:      $p\_current = partition$ 9
105:      $c\_current = getCurrentCoalition(eip_i)$ 10
116:      $c\_best = c\_current$ 11
127:     for  $c$  in  $p\_current \setminus c\_current$  do12
138:       if  $c$  not in  $h(eip_i)$  then13
149:          $c\_new = c \cup \{eip_i\}$ 14
1510:          $\Phi(c\_best) = getShapleyValue(c\_best, eip_i)$ 15
1611:          $\Phi(c\_new) = getShapleyValue(c\_new, eip_i)$ 16
1712:         if  $\Phi(c\_new) > \Phi(c\_best)$  then17
1813:            $c\_best = c\_new$ 18
1914:         end if19
2015:       end if20
2116:     end for21
2217:     if  $c\_best \neq c\_current$  then22
2318:       add set  $c\_current \setminus \{eip_i\}$  in  $h(eip_i)$ 23
2419:        $p\_best = (p\_current \setminus \{c\_current, c\_best \setminus \{eip_i\}\}) \cup (c\_current \setminus \{eip_i\}, c\_best)$ 24
2520:        $partition = p\_best$ 25
2621:     end if26
2722:     if  $c\_best == c\_current$  then27
2823:       break28
2924:     end if29
3025:   end while30
3126: end function31

```

²⁴ The function DCF accepts the variable eip_i and the global variable $partition$ used²⁴
²⁵to store the current coalition partition. The initial coalition partition is that every²⁵
²⁶EIP is a coalition. In DCFA, $A \setminus B$ represents that set A removes element B . The²⁶
²⁷algorithm has two important properties, which are convergence and Nash-stability²⁷
²⁸[41, 52].²⁸

³⁰**Theorem 2** Starting from any initial coalition structure, the DCFA algorithm³⁰
³¹always converges to a final partition.³¹

³³**Theorem 3** Any obtained final partition with algorithm DCFA is Nash-stable.³³

¹ Similar proof of theorem 2 and 3 can refer to the literature [53]. The property¹
² of Nash stability is a fundamental guarantee of convergence property. Moreover,²
³ introducing each member's visited history set makes the number of alternatives³
⁴ decrease monotonically to end with a stable coalition configuration.⁴

⁵ Experimental Evaluation⁵

⁶ Simulation Setup⁶

⁷ In this subsection, we consider a square region with $10km \times 10km$ located in Anning⁷
⁸ District, Lanzhou (China), as shown in Figure . The locations of EIPs and SPs are⁸
⁹ generated randomly in the region. We randomly generated 100 EIPs and 1000 SPs⁹
¹⁰ locations in total, and assumed that each SP is located at one location. We randomly¹⁰
¹¹ sampled with 8 EIPs, and 4 SPs from EIPs and SPs generated previously for the¹¹
¹² sake of clarity and analysis, i.e., $M = 8, S = 4$. We assume that the delay between¹²
¹³ an SP and an EIP is proportional to their distance. Suppose that the price for an SP¹³
¹⁴ to serve a request successfully is 2 to 3 per 100000. The physical infrastructure of¹⁴
¹⁵ each EIP has four identical edge nodes, and everyone has 40 CRBs, whose maximum¹⁵
¹⁶ power consumption W_k^{max} and idle power consumption W_k^{min} are 200w and 100w,¹⁶
¹⁷ respectively. The electricity price E_i to every EIP i is 0.0001/Wh. The penalty¹⁷
¹⁸ $P_{ij} = \beta R_{ji}$, where β is a factor. Here, let $\beta = 10$. The value of β ensures that each¹⁸
¹⁹ EIP prefers to allocate its CRBs for the services it hosts. Budget of each SP ranges¹⁹
²⁰ from 10 to 20, and we can normalize it to be 1 for generality. Parameters related to²⁰
²¹ services are shown in Table 1.²¹

²² Experimental Results²²

²³ According to the parameters set before, we get the valuation index v_{ij} that is²³
²⁴ each service i for all resources of EIP j , which is shown in Figure . Obviously,²⁴
²⁵ if we normalize v_{ij} corresponded to one CRB, it will have a generality. In this²⁵
²⁶ experiment, we assume that each EIP has an equal number of resources, such that²⁶
²⁷ the performance before and after normalization is equivalent. In Figure , each service²⁷
²⁸ corresponds to 8 EIPs evaluation indicators depicted by a bar graph. From Figure ,²⁸
²⁹ we can easily find that the optimal EIPs corresponding to services $\{1, 2, 3, 4\}$ are²⁹
³⁰ $\{5, 2, 3, 1\}$. The same conclusion can obtain from Figure to Figure . And from the³⁰
³¹ distribution map of SPs and EIPs (Figure), we inferred that the result obtained³¹
³² above is reasonable. For instance, we only consider one influencing factor – distance,³²
³³

¹SP 1 chooses the relatively farther EIP 5 instead of EIP 3. It is more reasonable¹
²to match SP 3 and EIP 3. Similarly, if SP 4 matches EIP 2, only can SP 2 match²
³EIP 7. However, EIP 7 is farther from SP 2 than EIP 2, and EIP 1 satisfies SP³
⁴enough. Overall, the matches are rational. Figure depicts the revenues of unit⁴
⁵cost, i.e., bang per buck $\frac{v_{ij}}{p_j}$, and Figure is the maximum bang per buck. Figure⁵
⁶and are similar in shape, but they represent utilities on different EIPs and the top⁶
⁷arrival rates supported by different EIPs, respectively.⁷

⁸ By solving the EG program, we get the Fisher market equilibrium prices of all⁸
⁹CRBs and unit CRB of EIPs under service providers' budgets, which described in⁹
¹⁰Figure . We set three budgets of SPs to analyze the impact of which on the market¹⁰
¹¹price; they are the benchmark budgets (10-20), double of the benchmark, and a half¹¹
¹²of the benchmark. The effects show in Figure .¹²

¹³ From Figure , all the prices increase twice as the budget of each SP is double. By¹³
¹⁴the same rule, all the prices reduce one-half, with each SP's budget reduced by half.¹⁴
¹⁵Hence, the value of the budget only affects the equilibrium prices by a scaling factor.¹⁵
¹⁶When fixing the budget, the prices act as a primary means to allocate resources¹⁶
¹⁷only.¹⁷

¹⁸ In the coalition formation algorithm, we vary the request intensity (arrival rate)¹⁸
¹⁹of services, and the corresponding required containers are changing with it. The¹⁹
²⁰detailed depiction of request intensity and corresponding containers show in Figure²⁰
²¹and Figure , where the red diamonds in these two figures represent the maximum²¹
²²request intensity and the corresponding required containers maximum supported²²
²³by EIPs selected.²³

²⁴ In Table 2, we present the coalitions formed, which correspond to different requests²⁴
²⁵of services, where $E[\cdot]$ denotes the arithmetic mean operator. In the first place,²⁵
²⁶depending on the columns *Average Arrival Rate* and *Average $E[\phi_i]$* , we have *Arrival*²⁶
²⁷*Rate 3 > Arrival Rate 4 > Arrival Rate 2 > Arrival Rate 1* and *Average $E[\phi_i]$* ²⁷
²⁸*2 > Average $E[\phi_i]$ 1 > Average $E[\phi_i]$ 4 > Average $E[\phi_i]$ 3*. As the requests increase,²⁸
²⁹we find that the benefits of cooperation increase first and then decrease. For one²⁹
³⁰thing, the increase in requests made them obtain more benefits, as more requests³⁰
³¹take up more resources. For another, as the requests increasing, the number of edge³¹
³²nodes switched off are decrease.³²

¹ We depict the utilities of different methods (fixed one-one contract-FOOC, coal-¹
²tion) in Figure , capturing that the EIPs' utilities obtained by cooperation are²
³markedly better than FOOC. And the arithmetic means of utilities of the coalition³
⁴method are also not less than FOOC. The experiments point out that coopera-⁴
⁵tion brings more significant benefits, and as the requests increase beyond a certain⁵
⁶amount, the advantages decrease gradually. ⁶

⁷ To analyze the effect of parameter β on the coalition formed and the average⁷
⁸utility of EIPs, we choose the arrival rate 2 as a case. The detailed results of the⁸
⁹experiment are shown in Table 3 and Figure , where $\beta = \{1, 5, 10, 15, 20, 30\}$ and⁹
¹⁰we can see that the average $\Psi\{i\}$ and $E[\phi_i]$ are decline linearly and monotonically¹⁰
¹¹with β increasing, which also can obtain from equations (24) and (25). Moreover, the¹¹
¹²gradient of the average $\Psi\{i\}$ and $E[\phi_i]$ are approximately equal. We find that as β ¹²
¹³increases, EIPs are more inclined to form a big coalition through further analysis. It¹³
¹⁴may be induced by that each EIP prefers to join an enormous coalition to apportion¹⁴
¹⁵more cost. ¹⁵

¹⁶ **Conclusion and future works** ¹⁷

¹⁸ In this work, we consider a scenario with a collection of SPs with different budgets¹⁸
¹⁹and several EIPs distributed in geographical locations. We aim to develop a price¹⁹
²⁰strategy to select EIPs with the most cost-effectiveness for each SP and get a stable²⁰
²¹coalition among EIPs selected with maximum EIP profits for different workloads. ²¹

²² For the scenario, we introduce the system architecture of MEC in an area. There²²
²³are three layers: the user layer, the aggregation layer, and the MEC layer. Towards²³
²⁴the first goal, we use the famous concept of General Equilibrium in Economics²⁴
²⁵as an effective solution. It produces a Market Equilibrium with Pareto-efficient and²⁵
²⁶fairness, etc., properties. We get the market equilibrium price and select the optimal²⁶
²⁷EIPs for each SP by solving the Eisenberg-Gale convex program. Aim at the second²⁷
²⁸purpose, depending on the price and EIPs selected, obtained by the first goal, we²⁸
²⁹use a mathematical model to maximize EIP's profit and an algorithm to form a²⁹
³⁰stable coalition. Numerical results demonstrate the effectiveness of the method. ³⁰

³¹ In future work, we will consider the net profit of the service provider and analyze³¹
³²the EIPs profit and the coalition under its market equilibrium price. ³²

1 Acknowledgements	1
2 We would like to thank Shuxu Zhao, Yangping Wang, the School of Electronic and Information Engineering,	2
3 Lanzhou Jiaotong University, Lanzhou, China, for providing the infrastructure and funding needed to carry out the	3
4 proposed research work.	3
4 Funding	4
5 This work was partially supported by National Natural Science Foundation of China (No. 6206020135), Key	5
6 Research and Development Program of Gansu Province (No. 20YF8GA123) and Innovation Star Project for	6
7 Outstanding Postgraduates of Gansu Province (No. 2021CXZX-557).	6
7 Abbreviations	7
8 MEC: Multi-access edge computing; EIP: Edge infrastructure provider; SP: Service provider; CRB: Computing	8
9 resource block; MBB: maximum bang-per-buck; EN: Edge nodes; IoT: Internet of things; DC: Cloud data center;	9
10 EG: Eisenberg-Gale; BS: Base station; AP: Access point; PoA: Point of aggregation; ME: Market equilibrium;	10
11 FOOC: Fixed one-one contract; MCEP: Market clearing equilibrium price.	10
11 Competing interests	11
12 The authors declare that they have no competing interests.	12
13 Authors' contributions	13
14 Xiaolong Wang: Formal analysis, Investigation, Methodology, Validation, Writing-original draft, Writing-review &	14
15 editing. Jianwu Dang: Conceptualization, Project administration, Funding acquisition, Writing - review & editing.	15
16 Shuxu Zhao: Funding acquisition, Methodology. Zhanping Zhang: Writing- review & editing. Yangping Wang:	16
17 Project administration, Funding acquisition. Zhanjun Hao: Conceptualization.	16
17 Author details	17
18 ¹ School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, 730070, China. ² College	17
19 of Computer Science & Engineering, Northwest Normal University, Lanzhou, 730070, China.	18
19 References	19
20 1. Hu, Y.C., Patel, M., Sabella, D., Sprecher, N., Young, V.: Mobile edge computing—a key technology towards	20
21 5g. ETSI white paper 11 (11), 1–16 (2015)	20
22 2. Mao, Y., You, C., Zhang, J., Huang, K., Letaief, K.B.: A survey on mobile edge computing: The	21
23 communication perspective. <i>IEEE Communications Surveys & Tutorials</i> 19 (4), 2322–2358 (2017).	21
24 doi:10.1109/COMST.2017.2745201	22
25 3. Abbas, N., Zhang, Y., Taherkordi, A., Skeie, T.: Mobile edge computing: A survey. <i>IEEE Internet of Things</i>	23
26 <i>Journal</i> 5 (1), 450–465 (2017). doi:10.1109/JIOT.2017.2750180	23
27 4. Zhang, T.: Data offloading in mobile edge computing: A coalition and pricing based approach. <i>IEEE Access</i> 6 ,	24
28 2760–2767 (2017). doi:10.1109/ACCESS.2017.2785265	24
29 5. Hassanali, M., Page, A., Soyata, T., Sharma, G., Aktas, M., Mateos, G., Kantarci, B., Andreescu, S.: Health	25
30 monitoring and management using internet-of-things (iot) sensing with cloud-based processing:	26
31 Opportunities and challenges. In: 2015 IEEE International Conference on Services Computing, pp. 285–292	26
32 (2015). doi:10.1109/SCC.2015.47. IEEE	27
33 6. Perera, C., Qin, Y., Estrella, J.C., Reiff-Marganiec, S., Vasilakos, A.V.: Fog computing for sustainable smart	28
34 cities: A survey. <i>ACM Computing Surveys (CSUR)</i> 50 (3), 1–43 (2017). doi:10.1145/3057266	28
35 7. Vasisht, D., Kapetanovic, Z., Won, J., Jin, X., Chandra, R., Sinha, S., Kapoor, A., Sudarshan, M., Stratman,	29
36 S.: Farmbeats: An iot platform for data-driven agriculture. In: 14th {USENIX} Symposium on Networked	30
37 Systems Design and Implementation ({NSDI} 17), pp. 515–529 (2017)	30
38 8. Byers, C.C.: Architectural imperatives for fog computing: Use cases, requirements, and architectural techniques	31
39 for fog-enabled iot networks. <i>IEEE Communications Magazine</i> 55 (8), 14–20 (2017).	31
40 doi:10.1109/MCOM.2017.1600885	32
41 9. Hong, C.-H., Varghese, B.: Resource management in fog/edge computing: a survey on architectures,	33
42 infrastructure, and algorithms. <i>ACM Computing Surveys (CSUR)</i> 52 (5), 1–37 (2019). doi:10.1145/3326066	33

10. Pei, Z., Piao, S., Souidi, M.E.H., Qadir, M.Z., Li, G.: Coalition formation for multi-agent pursuit based on neural network. *Journal of Intelligent & Robotic Systems* **95**(3), 887–899 (2019). doi:10.1007/s10846-018-0893-6
11. Hoefer, M., Vaz, D., Wagner, L.: Dynamics in matching and coalition formation games with structural constraints. *Artificial Intelligence* **262**, 222–247 (2018). doi:10.1016/j.artint.2018.06.004
12. Shi, W., Cao, J., Zhang, Q., Li, Y., Xu, L.: Edge computing: Vision and challenges. *IEEE internet of things journal* **3**(5), 637–646 (2016). doi:10.1109/JIOT.2016.2579198
13. Cao, J., Zhang, Q., Shi, W.: Challenges and opportunities in edge computing. In: *Edge Computing: A Primer*, pp. 59–70. Springer, ??? (2018). doi:10.1007/978-3-030-02083-5_5
14. Barbarossa, S., Sardellitti, S., Di Lorenzo, P.: Joint allocation of computation and communication resources in multiuser mobile cloud computing. In: *2013 IEEE 14th Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 26–30 (2013). doi:10.1109/SPAWC.2013.6612005. IEEE
15. Meng, S., Wang, Y., Miao, Z., Sun, K.: Joint optimization of wireless bandwidth and computing resource in cloudlet-based mobile cloud computing environment. *Peer-to-Peer Networking and Applications* **11**(3), 462–472 (2018). doi:10.1007/s12083-017-0544-x
16. Chen, M.-H., Dong, M., Liang, B.: Joint offloading decision and resource allocation for mobile cloud with computing access point. In: *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3516–3520 (2016). doi:10.1109/ICASSP.2016.7472331. IEEE
17. Dinh, T.Q., Tang, J., La, Q.D., Quek, T.Q.: Offloading in mobile edge computing: Task allocation and computational frequency scaling. *IEEE Transactions on Communications* **65**(8), 3571–3584 (2017). doi:10.1109/TCOMM.2017.2699660
18. Barbarossa, S., Di Lorenzo, P., Sardellitti, S.: Computation offloading strategies based on energy minimization under computational rate constraints. In: *2014 European Conference on Networks and Communications (EuCNC)*, pp. 1–5 (2014). doi:10.1109/EuCNC.2014.6882634. IEEE
19. Munoz, O., Pascual-Iserte, A., Vidal, J.: Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading. *IEEE Transactions on Vehicular Technology* **64**(10), 4738–4755 (2014). doi:10.1109/TVT.2014.2372852
20. Koshevoy, G., Suzuki, T., Talman, D.: Cooperative games with restricted formation of coalitions. *Discrete Applied Mathematics* **218**, 1–13 (2017). doi:10.1016/j.dam.2016.09.003
21. Chalkiadakis, G., Greco, G., Markakis, E.: Characteristic function games with restricted agent interactions: Core-stability and coalition structures. *Artificial Intelligence* **232**, 76–113 (2016). doi:10.1016/j.artint.2015.12.005
22. Kóczy, L.Á., Lauwers, L.: The coalition structure core is accessible. *Games and Economic Behavior* **48**(1), 86–93 (2004). doi:10.1016/j.geb.2003.06.006
23. Tang, L., He, S.: Multi-user computation offloading in mobile edge computing: A behavioral perspective. *IEEE Network* **32**(1), 48–53 (2018). doi:10.1109/MNET.2018.1700119
24. Wang, X., Liu, J.: A dynamic bargaining game with externalities. *Journal of Systems Science and Complexity* **31**(6), 1591–1602 (2018). doi:10.1007/s11424-018-7238-1
25. Tianze, L., Muqing, W., Min, Z., Wenxing, L.: An overhead-optimizing task scheduling strategy for ad-hoc based mobile edge computing. *IEEE Access* **5**, 5609–5622 (2017). doi:10.1109/ACCESS.2017.2678102
26. Zhang, Y., Wang, J., Liu, S., Qian, C.: Game theory based real-time shop floor scheduling strategy and method for cloud manufacturing. *International Journal of Intelligent Systems* **32**(4), 437–463 (2017). doi:10.1002/int.21868
27. Cao, X., Tang, G., Guo, D., Li, Y., Zhang, W.: Edge federation: Towards an integrated service provisioning model. *IEEE/ACM Transactions on Networking* **28**(3), 1116–1129 (2020)
28. Zhang, H., Xiao, Y., Bu, S., Niyato, D., Yu, F.R., Han, Z.: Computing resource allocation in three-tier iot fog networks: A joint optimization approach combining stackelberg game and matching. *IEEE Internet of Things Journal* **4**(5), 1204–1215 (2017). doi:10.1109/JIOT.2017.2688925
29. Brânzei, S., Chen, Y., Deng, X., Filos-Ratsikas, A., Frederiksen, S., Zhang, J.: The fisher market game: Equilibrium and welfare. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 28 (2014)
30. Maxfield, R.R.: General equilibrium and the theory of directed graphs. *Journal of Mathematical Economics*

- 1 27(1), 23–51 (1997). doi:10.1016/0304-4068(95)00763-6 1
- 2 31. Mas-Colell, A., Whinston, M.D., Green, J.R., et al.: Microeconomic Theory vol. 1. Oxford university press New 2
York, ??? (1995)
- 3 32. Liu, Z., Lin, M., Wierman, A., Low, S.H., Andrew, L.L.: Greening geographical load balancing. ACM 3
SIGMETRICS Performance Evaluation Review **39**(1), 193–204 (2011). doi:10.1145/2007116.2007139 4
- 4 33. Ardagna, D., Panicucci, B., Passacantando, M.: Generalized nash equilibria for the service provisioning problem 4
in cloud systems. IEEE Transactions on Services Computing **6**(4), 429–442 (2012). doi:10.1109/TSC.2012.14 5
- 5 34. Zhang, H., Xiao, Y., Bu, S., Niyato, D., Yu, F.R., Han, Z.: Computing resource allocation in three-tier iot fog 5
6 networks: A joint optimization approach combining stackelberg game and matching. IEEE Internet of Things 6
Journal **4**(5), 1204–1215 (2017). doi:10.1109/JIOT.2017.2688925 7
- 7 35. Tang, L., Chen, H.: Joint pricing and capacity planning in the iaaS cloud market. IEEE Transactions on Cloud 7
8 Computing **5**(1), 57–70 (2014). doi:10.1109/TCC.2014.2372811 8
- 9 36. Roughgarden, T.: Algorithmic game theory. Commun. ACM **53**(7), 78–86 (2010). 9
doi:10.1145/1785414.1785439
- 10 37. Devanur, N.R., Papadimitriou, C.H., Saberi, A., Vazirani, V.V.: Market equilibrium via a primal–dual algorithm 10
for a convex program. Journal of the ACM (JACM) **55**(5), 1–18 (2008). doi:10.1145/1411509.1411512 11
- 11 38. Boyd, S., Boyd, S.P., Vandenberghe, L.: Convex Optimization. Cambridge university press, ??? (2004). 11
doi:10.1017/CBO9780511804441 12
- 12 39. Nguyen, D.T., Le, L.B., Bhargava, V.: Price-based resource allocation for edge computing: A market 12
13 equilibrium approach. IEEE Transactions on Cloud Computing (2018). doi:10.1109/TCC.2018.2844379 13
- 14 40. Dreze, J.H., Greenberg, J.: Hedonic coalitions: Optimality and stability. Econometrica: Journal of the 14
Econometric Society, 987–1003 (1980). doi:10.2307/1912943
- 15 41. Bogomolnaia, A., Jackson, M.O.: The stability of hedonic coalition structures. Games and Economic Behavior 15
16 **38**(2), 201–230 (2002). doi:10.1006/game.2001.0877 16
- 17 42. Peleg, B., Sudhölter, P.: Introduction to the Theory of Cooperative Games vol. 34. Springer, ??? (2007) 17
- 18 43. Ma, R.T., Chiu, D.M., Lui, J.C., Misra, V., Rubenstein, D.: Internet economics: The use of shapley value for isp 17
18 settlement. IEEE/ACM Transactions on networking **18**(3), 775–787 (2010). doi:10.1109/TNET.2010.2049205 18
- 19 44. Kuhn, H.W., Tucker, A.W.: Contributions to the Theory of Games (AM-24), Volume I. Princeton University 19
Press, ??? (1951). doi:10.1515/9781400881727-002 19
- 20 45. Aumann, R.J., Dreze, J.H.: Cooperative games with coalition structures. International Journal of game theory 20
20 **3**(4), 217–237 (1974). doi:10.1007/BF01766876 20
- 21 46. Saad, W., Han, Z., Basar, T., Debbah, M., Hjørungnes, A.: Hedonic coalition formation for distributed task 21
allocation among wireless agents. IEEE Transactions on Mobile Computing **10**(9), 1327–1344 (2010). 22
doi:10.1109/TMC.2010.242 22
- 23 47. Anglano, C., Guazzone, M., Sereno, M.: Maximizing profit in green cellular networks through collaborative 23
games. Computer Networks **75**, 260–275 (2014). doi:10.1016/j.comnet.2014.10.003 23
- 24 48. Saad, W., Han, Z., Basar, T., Debbah, M., Hjørungnes, A.: A selfish approach to coalition formation among 24
unmanned air vehicles in wireless networks. In: 2009 International Conference on Game Theory for Networks, 25
pp. 259–267 (2009). doi:10.1109/GAMENETS.2009.5137409. IEEE 25
- 26 49. Kshemkalyani, A.D., Singhal, M.: Distributed Computing: Principles, Algorithms, and Systems. Cambridge 26
University Press, ??? (2011). doi:10.1017/CBO9780511805318 26
- 27 50. Weiss, G. (ed.): Multiagent Systems. 2nd Edition. MIT Press, ??? (2013) 27
- 28 51. Coulouris, Dollimore, J., Kindberg, T.: Distributed Systems: Concepts and Design (4th Edition) (International 28
Computer Science). Addison-Wesley Longman Publishing Co., Inc., USA (2005)
- 29 52. Anglano, C., Canonico, M., Castagno, P., Guazzone, M., Sereno, M.: Profit-aware coalition formation in fog 29
30 computing providers: A game-theoretic approach. Concurrency and Computation: Practice and Experience 30
32(21), 5220 (2020). doi:10.1002/cpe.5220
- 31 53. Guazzone, M., Anglano, C., Sereno, M.: A game-theoretic approach to coalition formation in green cloud 31
federations. In: 2014 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing, pp. 32
618–625 (2014). doi:10.1109/CCGrid.2014.37. IEEE 32
- 33 33 33

Figure 1 System Architecture**Figure 2** latency composition**Figure 3** Random distribution map of SPs and EIPs in Anning District, Lanzhou (China)**Figure 4** The evaluation indicator of Service to EIP**Figures****Tables****Table 1** Parameters related to services.

Service index	Service name	Latency threshold(ms)	Number of containers	Service rate per container
1	Healthcare	5	1	500
2	Accelerated Video	10	1	400
3	Virtual Reality	20	1	300
4	Web Game	30	1	200

Table 2 The Coalitions Formed with Different Arrival Rates

Arrival index	EIP	Request Intensity λ	Average arrival rate	N_{CRB}	$\Psi(\{i\})$	$E[\phi_i]$	Average $E[\phi_i]$	Coalition
1	5	32667.79	16114.82	66	8.7563	12.05	8.96	{1, 2, 3, 5}
	2	1095.622		5	1.4818	5.699		
	3	22626.079		92	8.1335	11.15		
	1	8069.79		50	4.5558	6.93		
2	5	41269.63	17879.925	83	8.7563	11.67	9.29	{5}, {1, 2, 3}
	2	23834.39		100	8.9011	12.00		
	3	2857.55		12	3.2534	6.98		
	1	3558.13		22	3.3476	6.50		
3	5	72456.64	18729.567	145	8.7563	16.72	7.026	{1, 2, 3, 5}
	2	1351.85		6	1.7802	4.33		
	3	1087.94		5	1.3539	3.98		
	1	21.839		1	0.1435	3.07		
4	5	16633.28	18610.117	34	9.9405	10.39	8.58	{5}, {1, 2, 3}
	2	50758.22		212	8.9011	13.09		
	3	1557.73		7	1.8995	4.35		
	1	5491.239		34	5.1799	6.47		

Figure 5 Revenue per unit cost(bang per buck)**Figure 6** Maximum revenue per unit cost(maximum bang per buck)**Table 3** The impact of β on coalition formed

β	Average $\Psi\{i\}$	Average $E[\phi_i]$	Coalition
1	7.4	10.42	$\{1, 5\}, \{2, 3\}$
5	6.807	9.97	$\{5\}, \{1, 2, 3\}$
10	6.065	9.29	$\{5\}, \{1, 2, 3\}$
15	5.32	8.657	$\{5\}, \{1, 2, 3\}$
20	4.58	8.03	$\{1, 2, 3, 5\}$
30	3.095	6.94	$\{1, 2, 3, 5\}$

Figure 7 The utilities on different EIPs**Figure 8** The maximum arrival rate for different EIPs**Figure 9** Market equilibrium prices of CRB on different EIP**Figure 10** The impact of different budgets on the market equilibrium price**Figure 11** different arrival rates**Figure 12** the number of CRBs corresponding to arrival rates**Figure 13** utilities of different methods for different arrival rate**Figure 14** The impact of β on average $\Psi\{i\}$ and $E[\phi_i]$

Figures

Figure 1

System Architecture

Figure 2

latency composition

Figure 3

Random distribution map of SPs and EIPs in Anning District, Lanzhou (China)

Figure 4

The evaluation indicator of Service to EIP

Figure 5

Revenue per unit cost(bang per buck)

Figure 6

Maximum revenue per unit cost(maximum bang per buck)

Figure 7

The utilities on different EIPs

Figure 8

The maximum arrival rate for different EIPs

Figure 9

Market equilibrium prices of CRB on different EIP

Figure 10

The impact of different budgets on the market equilibrium price

Figure 11

different arrival rates

Figure 12

the number of CRBs corresponding to arrival rates

Figure 13

utilities of different methods for different arrival rate

Figure 14

The impact of β on average $\Psi\{i\}$ and $E[\phi_i]$