Bernd Simon, Paul Adrian, Patrick Weber, Patrick Felka, Oliver Hinz, and Anja Klein
”A Bargaining Approach for Service Placement in Multi-Access Edge Computing with Information Asymmetries”, in *IEEE Transactions on Mobile Computing*, Early Access, 2025.

A Bargaining Approach for Service Placement in Multi-Access Edge Computing with Information Asymmetries

Bernd Simon*, Paul Adrian*, Patrick Weber†, Patrick Felka†, Oliver Hinz†, Anja Klein*
*Communications Engineering Lab, Technical University of Darmstadt, Germany, {b.simon, a.klein}@nt.tu-darmstadt.de
†Chair of Information Systems and Information Management,
Goethe University Frankfurt, {weber, felka, ohinz}@wiwi.uni-frankfurt.de

***Abstract*—Multi-access edge computing (MEC) refers to deploying computation resources, known as cloudlets or edge servers, near the edge of the mobile network. Services like augmented reality (AR) benefit from MEC by service placement, which refers to installing service-specific software and allocating resources on cloudlets. Service placement in MEC improves service quality and potentially reduces costs compared to centralized cloud computing approaches. The main stakeholders in MEC are infrastructure providers (IPs), who manage the MEC infrastructure, and service providers (SPs), who offer services to users. Both have unique technical and economic perspectives, such as resource demands, resource availability, and costs. Information asymmetries exist as only IPs have access to information about their resources, and only SPs have information about service usage and resource demands. This work addresses challenges of service placement in MEC from a multi-stakeholder, techno-economic perspective. We introduce a model including the stakeholders' technical and economic goals and information asymmetries. To solve this problem efficiently, we propose a multi-stakeholder bargaining mechanism, termed Nash Backward Induction with Linear Equilibrium Strategies (NBI-LES). In a case study with 544 users and 16 SPs, we achieve $79\,\%$ of the optimal reduction in traffic given by a centralized optimal service placement strategy.**

I. Introduction

Multi-Access Edge Computing (MEC) refers to deploying computational resources near the edge of the mobile network [1]. These resources are called cloudlets or edge servers. Cloudlets are small data centers within the one-hop communication range of the users' mobile devices. Recently, MEC has emerged as a new computing paradigm where centralized computing approaches such as cloud computing fall short [2]. Service placement in MEC refers to installing service-specific software on the cloudlet to enable a service to run on the cloudlet. In addition, each service placed on the cloudlet should be allocated computation and storage resources to process the users' service requests. In the following, when we refer to service placement, we include both the installation of the service-specific software and the allocation of computation and storage resources. Since the IP has deployed limited computation resources in the MEC network, the SPs must share these limited resources.

This work has been funded by the German Research Foundation (DFG) as a part of the projects B3, C5, and T2 within the Collaborative Research Center (CRC) 1053 - MAKI (Nr. 210487104) and has been supported by the BMBF project Open6GHub (Nr. 16KISK014).

The benefits of service placement in MEC lie in the improvement of service quality and the potential reduction of operational cost of networks [3] since the amount of communication from the edge through the network core is reduced, e.g., by more than 80% of bandwidth in a virtual reality context [4] and by up to 95% in an augmented reality (AR) context [5]. In addition, multiple types of services, including mobile gaming [6], smart home applications, or mobile crowdsensing [7], can benefit from service placement in MEC. Another important application of service placement in MEC is the caching of artificial intelligence (AI) models [8] and the inference of deep learning models at the edge of the wireless network [9].

A typical MEC system consists of users, service providers (SPs), infrastructure providers (IPs), and the cloud [10]. The users access a service, e.g., mobile gaming or AR application, by sending a service request. Each SP offers one of these services to users and is responsible for processing their service requests. To process the users' service requests, the SPs must use communication resources provided by the mobile network and computation resources provided by the IPs or the cloud. Thus, the SP performs the computation on a cloudlet in the MEC network or the cloud. The resources of the MEC network are provisioned by the IPs, where each IP may deploy one or more cloudlets at an access point [11]. For a service placement, i.e., processing the users' service requests on a cloudlet, we assume that the SP and the IP have to bargain about a price for the service placement. Using cloud resources, the SP has to pay a fixed price to provide the cloud resources.

To place a service on a cloudlet, both the IP and the SP must agree on the service placement. The SP has to pay for the service placement, and the IP must install its service-specific software on its cloudlet and provide computation and storage resources. Therefore, we consider the SPs and the IPs to be the primary decision-makers in the service placement problem. Each SP's goal is to provide its service to users at the lowest possible cost for the resources. The goal of each IP is to maximize its profit from the resource provisioning.

TABLE I: An overview of related work in the research area of service placement in MEC

Contributions \ Related Work	[12]–[17]	[18]	[19]	[20], [21]	[22]	[23]	[24]	[6]	[25], [26]	Our Approach
Techno-economic model of IP and SPs	-	✓	✓	✓	✓	✓	✓	✓	✓	✓
Joint solution for pricing and service placement	-	✓	-	✓	✓	✓	✓	✓	✓	✓
Fairness and efficiency considered	-	✓	✓	-	✓	-	-	✓	-	✓
Information asymmetries regarding resources	-	-	-	-	-	-	✓	-	✓	✓
Information asymmetries regarding cost factors	-	-	-	-	-	✓	✓	✓	-	✓
MEC model with limited resources	✓	-	✓	✓	✓	-	✓	-	✓	✓
Multiple SPs with heterogeneous services	✓	✓	✓	✓	✓	✓	✓	-	-	✓
Low computational complexity approach	✓	✓	✓	✓	✓	✓	-	✓	✓	✓

A. Research Challenges

The success of MEC requires a suitable service placement strategy to maximize both the IP's and the SP's revenue. Furthermore, novel bargaining mechanisms must consider fairness between the different stakeholders and ensure efficient use of the IPs' resources. However, designing a service placement strategy in MEC means overcoming three significant challenges:

1) Considering the techno-economic, multi-stakeholder perspective: The MEC network consists of multiple stakeholders, which are autonomous decision makers [1], [27]. The IP and the SPs act as individual rational stakeholders with their decision-making capabilities. Furthermore, each stakeholder has its individual technical and economic perspective, the so-called *techno-economic* perspective [27]. Considering service placement from the *technical* perspective requires careful analysis of the available resources and demands of all stakeholders. Considering service placement from the *economic* perspective requires analysis of the payments and costs of each stakeholder and finding a suitable pricing scheme. In the multi-stakeholder case, we must consider the *fairness* and *efficiency* of the provided solutions as discussed in [27] and [19]. *Fairness* of the proposed solution ensures that all stakeholders profit from using MEC; thus, participating is an incentive. Having an *efficient* solution ensures that the MEC network's communication and computation resources are utilized to achieve maximum performance. This techno-economic and multi-stakeholder perspective is important for a suitable service placement strategy in MEC networks [2].

2) Considering information asymmetries: Information asymmetries (IAs) occur as the SPs and the IP have different information available in a bargaining situation [6]. For example, the IP knows its available resources, specifically communication and computation resources, and the cost associated with using those resources. The SP knows about the expected service usage and the number of resources required to operate the service. Usually, the IPs and the SPs do not want to reveal this information. Therefore, in the multi-stakeholder case, finding an approach that allows bargaining without complete information to find an agreement and a suitable payment for service placement is essential.

3) Solving the combinatorial service placement problem given heterogeneous services and limited resources: Service types such as mobile gaming, AR, smart home, or mobile crowdsensing have heterogeneous communication and computation resource requirements. Additionally, the users' demand for each of these services dynamically changes over the course of a day. Furthermore, the IP has deployed only limited computation resources in the MEC network; thus, the SPs must share these limited resources. This results in an NP-hard combinatorial problem of finding the optimal service placement strategy, as shown in [17] and [28]. Therefore, service placement algorithms either rely on suboptimal heuristics to solve the service placement problem or use a low-complexity approximation of the problem to handle a large number of different services and adapt quickly to changing demands.

B. Related Works

Table I shows an overview of the related works. Previous works focusing on efficient service placement strategies from the perspective of a single stakeholder are [12]–[17]. In [12], the authors take the SPs' perspective and propose a centralized optimization framework for service placement in MEC networks to minimize the delay required for service request processing. The authors of [13] optimize the end-to-end delay in MEC using a low-complexity game-based approach to increase quality-of-service (QoS) by jointly considering the access network delay and the position of service placement while also considering the service placement cost. The authors of [14] take the IP's perspective and propose a low complexity centralized optimization approach using branch and bound and interior point methods to minimize the service placement cost while keeping a time constraint for all services. Efficient deployment of latency-critical virtual network function chains in MEC using a low complexity algorithm has been studied in [15] and [16]. In [12] and [14], assumptions include that the whole information of the MEC network, including all resources, services, and demands, is available to a central network orchestrator. In [17], the authors consider the IP's perspective and formulate a profit maximization problem for each cloudlet, which they solve using the interior point method. The service placement problem is solved for each cloudlet individually. Although the aforementioned works [12]–[16] and [17] significantly contributed to overcoming the challenge of efficient service placement, the multi-stakeholder perspective is missing in these works.

Related works [6] and [18]–[26] considered the multi-stakeholder perspective. In [18], the authors propose using Nash Bargaining to find an agreement for the service placement and pricing between the IP and the SPs. Assumptions include all the information about the resources and cost factors between the IP and the SPs. The authors of [19] propose to use Nash Bargaining for resource sharing between multiple SPs. Each SP has a utility depending on the amount of allocated resources. In [20], the authors propose a two-level optimization framework to optimize the IP's profit while minimizing the SPs' cost. Each SP has a limited budget for resources; therefore, setting the price too high may reduce the availability of services. In [21], the authors propose an auctioning-based mechanism to allocate the MEC resources of the IP to different heterogenous SPs. The SPs report their resource demand and QoS requirements to the IP and the IP determines a resource allocation and a price for the service placement. In [22], the authors model the service placement as an extended Fisher market. They present a convex optimization problem to find the market equilibrium of the proposed model of the extended Fisher market. In the aforementioned works [18]–[22], assumptions include that all stakeholders, namely IPs and SPs, share all their information regarding resources and cost factors. We argue this assumption is unrealistic, as SPs and IPs do not want to share their information.

The authors of [29] propose an admission control mechanism for an SP to accept or defer service requests coming from end users. The end users as well as the SP are considered to be stakeholders with their own utility functions, where the SP has to balance its revenue and the provided QoS. In [6] and [29] only one SP is considered.

Prior works that consider information asymmetries are [6] and [23]–[26]. In [6], the authors propose a multi-stakeholder bargaining procedure with partial information sharing. This approach is based on the NBS with IAs regarding the cost factors of the respective bargaining partner. In contrast to [6], multiple SPs are considered, and the limited resources of the IP are modeled. In [23], the authors propose a Stackelberg game with incomplete information to solve the IP's pricing problem and the SPs' placement decision. Assumptions include that the cloudlets have an infinitely long queue and unlimited resources. In the Stackelberg game, the authors model the SPs as passive price takers without their own bargaining power. The authors of [24] propose a double auction mechanism, where a centralized auctioneer takes bids for selling and buying resources. The centralized auctioneer acts as an individual stakeholder who profits from the difference between buying and selling prices. This adds additional costs for service placement in the MEC network. Furthermore, the authors did not consider the solution's fairness, and the proposed approach is highly computationally complex. In [25], the authors propose a two-stage dynamic game of incomplete information. In the first stage, the IP decides on the resource prices and service placement. In the second stage, the users decide whether to use the service provided. In [26], the authors propose a Bayesian optimization approach for the pricing of the SP. They select a price and a set of services for placement and learn the reward-maximizing placement strategy.

Known strategies from the literature [30] and [31] on how to overcome or reduce IAs are signaling, screening, monitoring, and information exchange. In this work, we focus on the IA reduction method of signaling, which refers to the voluntary sharing of information by the stakeholder possessing the information with the respective bargaining partner before agreeing on the payment and service placement, as discussed in [32].

As discussed, the prior works are limited in several ways. Firstly, the multi-stakeholder perspective is lacking in [6], [12], [14], [17], [23] and [26]. Secondly, the IAs are not considered in [18]–[20] and [22]. Thirdly, the perspective of efficiency and fairness is lacking in [20], [23]–[25] and [26]. To the best of our knowledge, we are the first to consider the service placement problem in MEC from a multi-stakeholder, techno-economic perspective, including IAs and considering limited resources.

C. Contributions

This work's main contribution is a novel multi-stakeholder bargaining mechanism that enables bargaining between the IP and the SPs under IAs. In the studied scenario, we consider one IP and multiple SPs with their individual utility functions consisting of technical and economic elements. We also consider IAs between the IP and the SPs. In our model, we adopt the simplifying assumption that bargaining occurs with a single IP. This approach is based on the premise that bargaining with multiple IPs can effectively be represented as simultaneous individual bargainings with each IP. Existing service placement approaches cannot handle the IAs between multiple stakeholders, resulting in degraded overall system performance. In particular, we propose a novel multi-stakeholder bargaining mechanism termed Nash Backward Induction with Linear Equilibrium Strategies (NBI-LES), whose goal is to find a fair and efficient solution to the service placement problem. In our NBI-LES approach, the IP and the SPs individually calculate their best LES strategy based on their information and belief about the bargaining partners' information. The main contributions of this work are summarized as follows.

- We model a bargaining situation between SPs offering many heterogeneous services and the IP. We include a fine-granular model of all relevant technical and economic attributes influencing service placement decisions. Furthermore, we discuss the different aspects of IAs between the SPs and the IP and provide a model that includes the stakeholders' beliefs about the respective bargaining partners.
- We propose a novel bargaining mechanism termed Nash Backward Induction with Linear Equilibrium Strategies (NBI-LES). We derive the optimal strategies for each stakeholder considering the given IAs and show that they are equivalent to the linear equilibrium strategies (LES), meaning no stakeholder would improve by changing its approach.
- To overcome the challenge of exponentially growing computational complexity of the service placement problem, we propose a discretization approach of the available

computation and storage resources into virtual machines (VMs) to achieve a solution with linearly growing complexity.
- We evaluate the performance of the proposed NBI-LES algorithm in a realistic case study. We analyze a data set containing the service usage of 544 users accessing various service types from 16 service providers. Furthermore, we use realistic models of the cost associated with data transfer and computation.

The rest of this paper has the following structure. In Section II, we introduce the MEC system model with the relevant stakeholders, and in Section III, we present the proposed NBI-LES algorithm. The evaluation of the proposed NBI-LES algorithm in an extensive case study follows in Section IV, and finally, Section V concludes the paper.

II. System model

A. Overview

Figure 1 shows the system model. The two relevant stakeholders are the IP and the SPs, the main decision-makers in MEC's service placement. The IP owns and operates the resources at the edge, i.e., the base station (BS), the cloudlet, and the backhaul network. The backhaul network connects the edge of the network, i.e., the BS and the cloudlet, to the core network. The BS uses a wireless link to transfer data to the users' devices. Further, the BS hosts a cloudlet, a computation resource close to the users, offering low-latency access to the services placed on it. In contrast to the centralized cloud, edge computing devices, such as cloudlets, have limited computation resources [33]. We consider N different SPs that each offer one service to the users. Each SP requires communication, computation, and storage resources in the network to offer its service. The computation and storage resources are available either in the centralized cloud or in the cloudlet provided by the IP. The users pay for the access to the service.

The IP bargains with each SP individually about whether or not to place the service on the cloudlet and which price has to be paid for the service placement. This individual bargaining protocol allows the IP as well as the heterogeneous SPs to make individual agreements, which improves the efficiency of the resource allocation.

We assume a non-cooperative setting, as described in [34], where SPs compete and bargain individually rather than forming groups or coalitions, which reflects the market dynamics in many practical scenarios. This means that each SP individually performs a one-to-one bargaining with the IP, i.e., SP 1 negotiates with the IP, then SP 2 negotiates with the IP until finally SP N and the IP bargain about the service placement. In our terminology, SP 1 denotes the SP that begins bargaining with the IP, while SP n represents the SP in the n-th bargaining step.

Each bargaining has two possible outcomes: a service placement agreement or no agreement. An agreement results in the service being deployed on the cloudlet and the SP paying

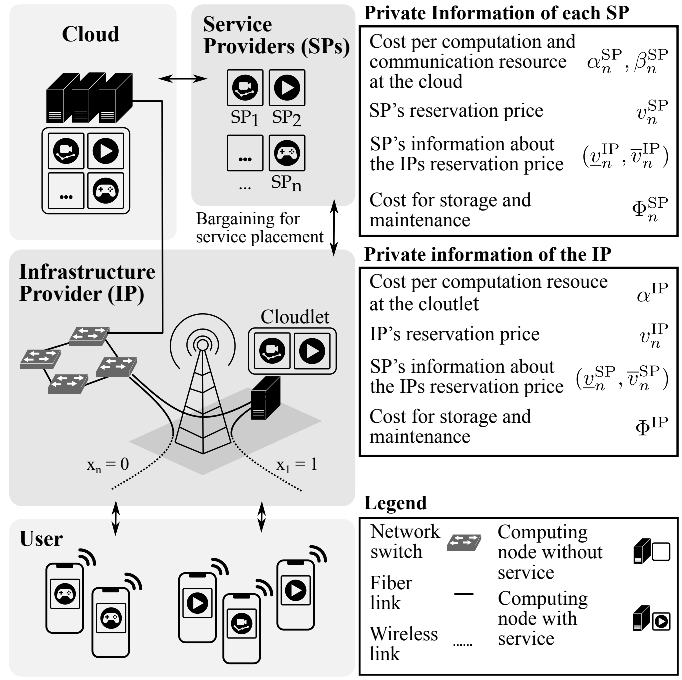


Fig. 1: Overview of the system model

the IP. When the IP places the service of SP n on the cloudlet, we denote this by an indicator variable $x_n = 1$. In this case, the IP receives a payment k_n from SP n. In the case of no agreement, the SP continues to rely on cloud computing, which we denote by $x_n = 0$. When the IP does not place the service on the cloudlet, there is no payment from the SP to the IP ($k_n = 0$). To simplify the notation, we introduce the *service placement profile* $\boldsymbol{x} = (x_1, \ldots, x_N)$ and the *payment profile* $\boldsymbol{k} = (k_1, \ldots, k_N)$. Table II shows the mathematical symbols introduced in the system model.

B. Service Providers

We consider a set $\mathcal{N} = \{1, \ldots, N\}$ of SPs, each offering one service to its users. The users of SP n access the service by sending service requests and paying the SP n for access to the service. To process these service requests, the SP needs to use communication, computation, and storage resources in the network. The services are heterogeneous, so the communication and computation requirements are different for each service request. For each service, we consider all users' communication, computation, and storage resource demand. These demands are individually aggregated over all the users' service requests. SP n denotes the aggregated communication demand as D_n^{SP} and the aggregated computation demand as X_n^{SP}. Furthermore, we denote the sum of all users' payments as $k_n^{\mathrm{Users,SP}}$.

Depending on the bargaining result, there are two possibilities to provision these resources: The provisioning of resources occurs either directly at the network edge using MEC in the case of an agreement or the centralized cloud through cloud computing in the absence of an agreement. In the following, we discuss the two different cases: 1) the SP uses cloud resources, and 2) the SP uses service placement in MEC.

1) Cloud Computing: In this paragraph, we discuss the case of the SPs offering their service using cloud resources. SP n needs to pay for using the communication, computation and storage resources in the cloud. In particular, we assume that SP n has negotiated a fixed price per resource in the cloud. We model the communication cost between the users' mobile devices and the cloud using a cost factor β_n^{SP} for each unit of data transmitted. Additionally, the SP must pay α_n^{SP} for each computation resource required in the cloud. We denote the storage cost in the cloud for the service by SP n by Φ_n^{SP}. In the case of cloud computing, the utility of SP n is

$$\begin{aligned} U_n^{\mathrm{SP}}(x_n=0,k_n) = & k_n^{\mathrm{Users,SP}} - \beta_n^{\mathrm{SP}} D_n^{\mathrm{SP}} - \alpha_n^{\mathrm{SP}} X_n^{\mathrm{SP}} \\ & - \Phi_n^{\mathrm{SP}}, \end{aligned} \tag{1}$$

which is the difference between the aggregated payments $k_n^{\mathrm{Users,SP}}$ of its users and the total cost for communication, computation and storage.

2) Service placement in Multi-Access Edge Computing: In this case, the SP n has successfully negotiated with the IP to place its service on the cloudlet. The IP installs the service-specific software of SP n on the cloudlet and provisions the necessary computation and storage resources to process the users' service requests of the service provided by SP n on the cloudlet. This significantly reduces the latency, improving the users' Quality-of-Service (QoS). To account for the improved QoS, we assume that users are potentially willing to pay more for the service. For this, we introduce an increase in payment L_n^{SP}. Note that L_n^{SP} can be zero if the users are unwilling to pay more for an increased QoS.

The SP has to pay k_n monetary units to the IP for the placement. Determining k_n is part of the bargaining procedure, discussed in Section III. Despite the utilization of MEC, there remains a need to communicate with the cloud and execute computations at the cloud, e.g., for synchronization with a database [6]. We denote $\widetilde{D}_n^{\mathrm{SP}}$ as the communication demand to the cloud and $\widetilde{X}_n^{\mathrm{SP}}$ as the computation demand in the cloud. The utility of the SP n using MEC is

$$\begin{aligned} U_n^{\mathrm{SP}}(x_n=1,k_n) = & k_n^{\mathrm{Users,SP}} + L_n^{\mathrm{SP}} - \beta_n^{\mathrm{SP}} \widetilde{D}_n^{\mathrm{SP}} \\ & - \alpha_n^{\mathrm{SP}} \widetilde{X}_n^{\mathrm{SP}} - k_n, \end{aligned} \tag{2}$$

which is the difference between the users' payments, the remaining cost in the cloud, and the payment to the IP for service placement.

The linear utility function (2) is based on two assumptions: Firstly, no saturation effects are considered, meaning the utility of each SP is linearly dependent on the cost parameters $\beta_n^{\mathrm{SP}}, \alpha_n^{\mathrm{SP}}$, the communication demand $\widetilde{D}_n^{\mathrm{SP}}$ and computation demand $\widetilde{X}_n^{\mathrm{SP}}$, without diminishing returns from increased resource demands or higher costs. Secondly, we assume risk neutrality, meaning that the SPs aim to maximize expected profit without incorporating terms related to risk, which are typically non-linear.

The payoff which SP n gains in case of a service placement is

$$\begin{aligned} P_n^{\mathrm{SP}}(k_n) = & U_n^{\mathrm{SP}}(x_n=1,k_n) - U_n^{\mathrm{SP}}(x_n=0,k_n) \\ = & L_n^{\mathrm{SP}} + \alpha_n^{\mathrm{SP}}(X_n^{\mathrm{SP}} - \widetilde{X}_n^{\mathrm{SP}}) \\ & + \beta_n^{\mathrm{SP}}(D_n^{\mathrm{SP}} - \widetilde{D}_n^{\mathrm{SP}}) - k_n \end{aligned} \tag{3}$$

which is the difference between the utility of edge computing and the utility in the case of cloud computing.

We define the reservation price v_n^{SP} of SP n as the highest payment the SP is willing to make for service placement. This reservation price appears when the payoff from (3) is $P_n^{\mathrm{SP}}(v_n^{\mathrm{SP}}) = 0$. This can be determined as

$$\begin{aligned} v_n^{\mathrm{SP}} = & L_n^{\mathrm{SP}} + \alpha_n^{\mathrm{SP}}(X_n^{\mathrm{SP}} - \widetilde{X}_n^{\mathrm{SP}}) \\ & + \beta_n^{\mathrm{SP}}(D_n^{\mathrm{SP}} - \widetilde{D}_n^{\mathrm{SP}}). \end{aligned} \tag{4}$$

We assume that each of the SPs is individually rational, i.e., they only agree to a payment k_n which results in a positive payoff $P_n^{\mathrm{SP}}(k_n) \geq 0$ from (3). In the bargaining with the IP, SP n is therefore never willing to make a payment $k_n > v_n^{\mathrm{SP}}$ which is larger than its reservation price v_n^{SP}. Therefore v_n^{SP} is the maximum price that SP n is willing to pay for the service placement.

C. Infrastructure Provider

The IP owns and operates the cloudlet, including the backhaul network between the BS and the cloud. To derive the utility function of the IP, we first consider the negotiation with one SP. When negotiating with SP n, there are two cases: Either the SP n provides the service using cloud computing, i.e., not using the cloudlet of the IP, or the IP provides the service of SP n using MEC, i.e., relying on the IP's infrastructure. In the following, we discuss those two different cases.

1) Cloud Computing of SP n: In the case of cloud computing of SP n, indicated by $x_n = 0$, there is no payment from SP n to the IP. The IP incurs a given cost β^{IP} for each data unit transferred between the users and the core network on the IP's backhaul network. Therefore, the utility is

$$U_n^{\mathrm{IP}}(x_n=0,k_n) = -\beta^{\mathrm{IP}} D_n^{\mathrm{SP}}, \tag{5}$$

which is the cost for data transfer from the user to the cloud resources.

2) Multi-Access Edge Computing of SP n: In the case of edge computing of SP n, i.e., $x_n = 1$, the SP n and the IP must agree on a payment k_n for the usage of the cloudlet. When using the cloudlet, the IP's cost comprises computation, communication and storage costs. Computation on the cloudlet incurs a cost of α^{IP} for each computation resource. Furthermore, for each communication resource required between the users and the cloud, there is a cost of β^{IP}. The infrastructure cost for storage is Φ_n^{IP}. The utility of the IP in the case of edge computing is

$$U_n^{\mathrm{IP}}(x_n=1,k_n) = k_n - \alpha^{\mathrm{IP}} X_n^{\mathrm{SP}} - \beta^{\mathrm{IP}} \widetilde{D}_n^{\mathrm{SP}} - \Phi_n^{\mathrm{IP}}, \tag{6}$$

TABLE II: Overview of the notation and the information asymmetries between the IP and the SPs

	Symbol	Description		Symbol	Description
Information of the Infrastructure Provider	$U^{\mathrm{IP}}(\boldsymbol{x}, \boldsymbol{k})$	Utility function of the IP	**Information of Service Provider n**	$U_n^{\mathrm{SP}}(x_n, k_n)$	Utility function of SP n
	X^{IP}	Available computation resources at the cloudlet		X_n^{SP}	Computation resource demand of SP n on the cloudlet
	x_n	Indicator variable $x_n \in \{0,1\}$ whether the service from SP n was placed		$\widetilde{X}_n^{\mathrm{SP}}$	Computation resource demand of SP n on the cloud
	$\boldsymbol{x} = (x_1, \dots, x_N)$	Service placement profile		D_n^{SP}	Communication resource demand between the BS of the IP and the cloud caused by SP n's users
	k_n	Payment from SP n to the IP		$\widetilde{D}_n^{\mathrm{SP}}$	Communication resource demand between the BS of the IP and the cloud caused by SP n's users in case of service placement
	$\boldsymbol{k} = (k_1, \dots, k_N)$	Payment profile		L_n^{SP}	Additional utility by service placement, e.g., by increased QoS for the users
				$k_n^{\mathrm{Users,SP}}$	Payments from the users to SP n for the service usage in the considered time interval
	α^{IP}	Cost per computation resource at the cloudlet		α_n^{SP}	Cost per computation resource at the cloud
	β^{IP}	Cost per bit for communication on the backhaul network		β_n^{SP}	Cost for communication between user and the cloud
	Φ_n^{IP}	Infrastructure cost per bit for storage and maintenance cost		Φ_n^{SP}	Cost per bit for storage and maintenance cost of the service provided by SP n
	v_n^{IP}	IP's reservation price when bargaining with the SP n		v_n^{SP}	SP's reservation price when bargaining with the IP
	$f_n^{\mathrm{IP}}(v_n^{\mathrm{SP}})$	IP's belief of the SP's reservation price		$f_n^{\mathrm{SP}}(v_n^{\mathrm{IP}})$	SP's belief of the IP's reservation price

i.e., the difference between the payment k_n of the SP n and the cost for computation, communication and storage resources. Analogous to the SPs, we also assume a linear utility function (6) for the IP, without saturation effects and under the assumption of risk neutrality.

Analogous to the payoff (3) of the SPs, we define the payoff of the IP for an agreement with SP n as the difference between the utility for edge computing (6) and for cloud computing (5). The payoff of the IP in case of an agreement with SP n is

$$\begin{aligned} P_n^{\mathrm{IP}}(k_n) &= U_n^{\mathrm{IP}}(x_n = 1) - U_n^{\mathrm{IP}}(x_n = 0) \\ &= k_n - \alpha^{\mathrm{IP}} X_n^{\mathrm{SP}} - \beta^{\mathrm{IP}}(\widetilde{D}_n^{\mathrm{SP}} - D_n^{\mathrm{SP}}) - \Phi_n^{\mathrm{IP}}. \end{aligned} \tag{7}$$

From the payoff, we can derive the reservation price

$$v_n^{\mathrm{IP}} = \alpha^{\mathrm{IP}} X_n^{\mathrm{SP}} + \beta^{\mathrm{IP}}(\widetilde{D}_n^{\mathrm{SP}} - D_n^{\mathrm{SP}}) + \Phi_n^{\mathrm{IP}} \tag{8}$$

which is the cost for the resources required for the service placement of SP n. We refer to v_n^{IP} as the IP's *physical* cost of the service placement, as (8) includes all the costs associated with the provisioning of the computation, communication and storage resources.

By X^{IP}, we denote the IP's limited computation resources at the cloudlet. To model the limited resources, we introduce the constraint

$$\sum_{n=1}^{N} x_n X_n^{\mathrm{SP}} \leq X^{\mathrm{IP}} \tag{9}$$

for the computation resources at the cloudlet.

The total utility of the IP is the sum

$$U^{\mathrm{IP}}(\boldsymbol{x}, \boldsymbol{k}) = \sum_{n=1}^{N} U_n^{\mathrm{IP}}(x_n, k_n) \tag{10}$$

over all potential agreements with all N SPs.

D. Information asymmetries

We consider different types of IAs in the scenario: IAs regarding costs and utility functions and IAs regarding the availability of resources. In Table II, we highlight these IAs by displaying the information of the IP on the left side and the information of the SPs on the right side.

1) Information asymmetries regarding the available resources: The IP knows the amount X^{IP} of (its own) available computation resources, which is unknown to the SPs. Furthermore, each SP does not know the number $N-1$ of other SPs competing for the resources. The IP knows the number N of its bargaining partners. This is an advantage in the bargaining procedure, as the IP knows how limited the resources are.

2) Information asymmetries regarding costs and utility functions: The IP and the SPs do not share information about their cost factors and utility functions, as they contain sensitive business model information. If, for example, the IP knew the exact cost factor and utility function of an SP, the IP could price its resources exactly at the reservation price of the SP and maximize the IP payoff. Therefore, no bargaining would occur, and the SP would not receive any payoff. Thus, we assume that the IP and the SPs do not know the cost factors and utility functions (1), (2), (5) and (6) of their bargaining partners. Therefore, the respective bargaining partners' payoffs (3) and (7) are unknown. However, this information is essential for settling the payment k_n between the SP n and the IP, in a manner reflecting optimal payoffs for the IP *and* the SP. For example, a high reservation price v_n^{SP} of SP n would mean that the IP could demand a higher payment for the service placement. Although SPs and IP do not know the exact reservation prices of each other, the SPs and the IP can obtain some information about their bargaining partner's reservation price, e.g., using publicly available information sources [6]. To model these IAs, we assume that both the SPs and the IP can calculate a probabilistic model of the bargaining partner's reservation price. A common approach in bargaining theory to model the belief over the bargaining partner's payoff is a uniform distribution between a lower bound and an upper bound [35], [36].

Each SP n can model its belief over the IP's reservation price v_n^{IP} using a probability density function

$$f_n^{\mathrm{SP}}(v_n^{\mathrm{IP}}) = \begin{cases} \frac{1}{\overline{v}_n^{\mathrm{IP}} - \underline{v}_n^{\mathrm{IP}}}, & \text{for } \underline{v}_n^{\mathrm{IP}} < v_n^{\mathrm{IP}} < \overline{v}_n^{\mathrm{IP}} \\ 0, & \text{else,} \end{cases} \tag{11}$$

which is a uniform distribution between a lower bound $\underline{v}_n^{\mathrm{SP}}$ and an upper bound $\overline{v}_n^{\mathrm{SP}}$. The difference between $\underline{v}_n^{\mathrm{SP}}$ and $\overline{v}_n^{\mathrm{SP}}$ is inversely proportional to the amount of information available to SP n. In a practical scenario, the SP has to estimate the lower bound $\underline{v}_n^{\mathrm{IP}}$, which is the minimum price the IP would accept for service placement. The SP can obtain the lower bound $\underline{v}_n^{\mathrm{IP}}$, e.g., by an estimation of the costs for the additional hardware and energy of the IP for service placement as described in [37]. The upper bound $\overline{v}_n^{\mathrm{IP}}$ of the IP's reservation price can be obtained, e.g., by using prices for service placement [38].

Analogously, the IP models its belief about each SP's reservation price as

$$f_n^{\mathrm{IP}}(v_n^{\mathrm{SP}}) = \begin{cases} \frac{1}{\overline{v}_n^{\mathrm{SP}} - \underline{v}_n^{\mathrm{SP}}}, & \text{for } \underline{v}_n^{\mathrm{SP}} < v_n^{\mathrm{SP}} < \overline{v}_n^{\mathrm{SP}} \\ 0, & \text{else.} \end{cases} \tag{12}$$

Similar to the case of the SP, the IP has to estimate its upper bound $\overline{v}_n^{\mathrm{SP}}$ by using public price lists of cloud providers, e.g., [39]. Its lower bound $\underline{v}_n^{\mathrm{SP}}$ can be obtained by estimating the reduction in cloud and backhaul cost of the SP, e.g., as discussed in [40].

Note that the SP and the IP cannot employ learning approaches, as observing the true values of the bargaining partner's reservation prices is impossible.

E. Social Welfare

We define the social welfare as the sum of the utilities of all SPs and the IP:

$$\begin{aligned} \Psi(\boldsymbol{x}, \boldsymbol{k}) &= \sum_{n=1}^{N} U_n^{\mathrm{SP}}(x_n, k_n) + U^{\mathrm{IP}}(\boldsymbol{x}, \boldsymbol{k}) \\ &= \sum_{n=1}^{N} x_n (v_n^{\mathrm{SP}} + v_n^{\mathrm{IP}}). \end{aligned} \tag{13}$$

The social welfare $\Psi(\boldsymbol{x}) = \Psi(\boldsymbol{x}, \boldsymbol{k})$ is therefore independent of the payment profile $\boldsymbol{k}$ [41]. Consequently, the social welfare $\Psi(\boldsymbol{x})$ measures the benefit for the whole network based on the chosen service placement profile $\boldsymbol{x}$.

The payment profile $\boldsymbol{k}$ determines the distribution of the payoff of the service placement between the IP and the SPs.

F. Fairness

Another important measure of the provided solution is fairness. SPs may not place their service at the edge if they do not benefit from it. Therefore, it is an important incentive that the benefits of MEC are shared in a *fair* manner between all stakeholders. A common approach to measure fairness is the Jain's index, e.g., as discussed in [19]. The Jain's index for the payoffs of all SPs which have their service placed is given by

$$\mathcal{J}(\boldsymbol{x}, \boldsymbol{k}) = \frac{(\sum_{n=1}^{N} P_n^{\mathrm{SP}}(k_n))^2}{(\sum_{n=1}^{N} P_n^{\mathrm{SP}}(k_n)^2)(\sum_{n=1}^{N} x_n)}, \tag{14}$$

which is the quotient of the squared mean value of the payoffs and the expected value of the squared payoff. A Jain's index of $\mathcal{J}(\boldsymbol{x}, \boldsymbol{k}) = 1$ indicates a fair service placement and pricing, i.e., all stakeholders have the same payoffs. The least fair solution is indicated by $\mathcal{J}(\boldsymbol{x}, \boldsymbol{k}) = \dfrac{1}{(\sum_{n=1}^{N} x_n)}$.

G. Problem Formulation

We argue that a formulation of the service placement problem as an optimization problem, e.g., as proposed in [42] and [43], is not realistic in a practical scenario, as we consider the SPs and the IP to act as selfish stakeholders with their individual decision-making capabilities. Furthermore, a centralized optimization approach requires a central entity with full knowledge of all SPs and the IP.

Moreover, the goal is not only to maximize efficiency but also to enhance fairness [19]. The Nash Bargaining Solution (NBS) [44] provides a fair and efficient outcome [45]. However, we argue that the NBS cannot be calculated directly, e.g., as proposed in [18], as no central entity with complete information can calculate the NBS.

1) One-to-one bargaining with information asymmetries: When only one SP n exists, we can describe the situation as a buyer-seller problem with IAs. In this case, the SP n acts as buyer, paying for the use of the available computation resources X^{IP} on the cloudlet, while the IP acts as seller, provisioning the resources to the SP n. The IP aims to maximize the payment k_n and will never accept any price below its reservation price v_n^{IP}. The SP aims to minimize the payment k_n and will never accept any price higher than its reservation price v_n^{SP}. As a result, an agreed-upon price between the two parties must lie in the interval $k_n \in (v_n^{\mathrm{IP}}, v_n^{\mathrm{SP}})$. No efficient placement exists in the case of $v_n^{\mathrm{IP}} > v_n^{\mathrm{SP}}$ [6]. We define the *one-to-one* bargaining problem between one IP and one SP as

$$\mathbf{P1}: \quad \underset{x_n, k_n}{\arg\max} \quad (U_n^{\mathrm{SP}}(x_n, k_n) - U_n^{\mathrm{SP}}(0,0)) \tag{15}$$

$$\cdot (U_n^{\mathrm{IP}}(x_n, k_n) - U_n^{\mathrm{IP}}(0,0))$$

$$\text{s.t.} \quad x_n X_n^{\mathrm{SP}} \leq X^{\mathrm{IP}} \tag{16}$$

which is the NBS with the computing resource constraint (16). The solution of (15) cannot be calculated directly because of the IAs.

2) One-to-many bargaining with information asymmetries: In the case of $N > 1$ SPs requesting a service placement on the IP's cloudlet, we must consider the one-to-many bargaining. The SPs have reservation prices v_n^{SP}, and the IP has reservation prices v_n^{IP} for the bargainings. In a scenario with *unlimited* resources, the case would simplify to N independent one-to-one bargainings as discussed in [6]. As we assume limited cloudlet resources given by (9), we need to account for that in our bargaining mechanism. Due to the scarcity of resources, the cloudlet cannot host every service. This corresponds to an opportunity cost that we must reflect in the reservation price. Therefore, we formulate the reservation price for the IP as the sum

$$\tilde{v}_n^{\mathrm{IP}} = v_n^{\mathrm{IP}} + v_n^{\mathrm{IP,opp}} \tag{17}$$

of the physical cost v_n^{IP} (8) and opportunity cost $v_n^{\mathrm{IP,opp}}$ associated with the service placement.

III. Proposed Nash Backward Induction with Linear Equilibrium Strategies Algorithm

This section presents our proposed bargaining mechanism for solving the service placement problem. The two main challenges we need to overcome are IAs and limited resources. In the bargaining situation with N SPs, the service placement considering the limited resources of the IP is challenging as the IP has opportunity costs that each agreement with an SP induces. Unlike [23] and [25], which model the IP as a price-setting leader and the SPs as price-taking followers in a Stackelberg game, we propose a bargaining framework between the SPs and the IP. This approach more accurately reflects the bargaining powers of both the IP and the SPs.

A. Sequential bargaining mechanism

Our proposed bargaining mechanism is a sequential bargaining approach, i.e., the IP bargains with each SP individually. We depict the sequential bargaining mechanism in Algorithm 1. Initially, each SP reports its computation resources demand $\widetilde{X}_n^{\mathrm{SP}}$ to the IP (line 3). After all N SPs have signaled whether they want to participate in the bargaining, the IP calculates its optimal strategy (lines 4-9). Simultaneously, the SPs calculate their optimal offer strategies (line 12). Afterward, the individual one-to-one bargainings start (lines 14-21). There are N bargaining rounds; each bargaining is between one SP $n \in \{1, \ldots, N\}$ and the IP. Due to potential agreements in previous rounds, the amount of available resources X^{IP} might decrease each bargaining round. We denote the available resources in bargaining round n as X_n^{IP}.

For each bargaining round, we propose to use a double auction mechanism, as described in [46] and [47]. In this mechanism, the SP n and IP simultaneously suggest a price for the service placement. The IP makes an offer k_n^{IP} and the SP n makes an offer k_n^{SP} in bargaining round n. If $k_n^{\mathrm{SP}} > k_n^{\mathrm{IP}}$, i.e., the SP is willing to pay more than the IP requests, then the service of SP n is placed on the cloudlet and the final agreed payment

$$k_n = \frac{k_n^{\mathrm{IP}} + k_n^{\mathrm{SP}}}{2} \tag{18}$$

is the average of the two offers. If $k_n^{\mathrm{SP}} < k_n^{\mathrm{IP}}$, i.e., the SP is willing to pay less than the IP requests, then no agreement is reached. Either way, the bargaining concludes with a reveal of both offer prices. This mechanism is considered fair as both offers of IP and SP receive equal weights.

Algorithm 1 Proposed NBI-LES algorithm

1: **Input:** Reservation prices $v_{\mathrm{SP},n}$ and $v_{\mathrm{IP,physical},n}$
2: **Input:** Belief about the $f_n^{\mathrm{IP}}(v_n^{\mathrm{SP}})$, $f_n^{\mathrm{SP}}(v_n^{\mathrm{IP}}) \quad \forall n$
3: All N SPs report X_N^{SP} to the IP
4: ▷ *The IP calculates its optimal offers*
5: The IP computes $\pi_N^{\mathrm{IP}}(X_N^{\mathrm{IP}})$ ▷ Eq. (20)
6: **for** $n = N-1, ..., 1$ **do**
7:     The IP computes $v_n^{\mathrm{IP}}(X_n^{\mathrm{IP}})$ ▷ Eq. (22)
8:     The IP computes $\pi_n^{\mathrm{IP,agg}}(X_n^{\mathrm{IP}})$ ▷ Eq. (24)
9:     The IP computes k_n^{IP} according to the LES ▷ Eq. (25)
10: **end for**
11: ▷ *Each SP calculates its optimal offer*
12: Each SP computes k_n^{SP} according to the LES ▷ Eq.(27)
13: ▷ *Sequential bargaining*
14: **for** $n = 1, ..., N$ **do**
15:     SP n makes the offer k_n^{SP}, the IP k_n^{IP}.
16:     **if** $k_n^{\mathrm{IP}} \leq k_n^{\mathrm{SP}}$ **then**
17:         $x_n = 1$, $k_n = \frac{1}{2}(k_n^{\mathrm{IP}} + k_n^{\mathrm{SP}})$ ▷ Eq. (18)
18:         $X_{n+1}^{\mathrm{IP}} = X_n^{\mathrm{IP}} - X_n^{\mathrm{SP}}$
19:     **else**
20:         $x_n = 0, k_n = 0$
21:         $X_{n+1}^{\mathrm{IP}} = X_n^{\mathrm{IP}}$
22:     **end if**
23: **end for**
24: **return** service placement profile $\boldsymbol{x}$, payment profile $\boldsymbol{k}$

The challenge is now to calculate the optimal offer k_n^{IP} of the IP and the optimal offer k_n^{SP} of the SPs. Logically, the IP and each SP will optimize the respective offers to maximize their expected payoff. Any stakeholder's optimal offer strategy is dependent not only on its own reservation price but also on the reservation price and offer strategy of the other parties. Furthermore, due to IAs, the optimal strategy depends on the belief about the others' reservation prices.

B. The strategy of the infrastructure provider

The following section will discuss IP's strategy for calculating reservation prices and optimal offers. Afterwards, we discuss the SPs' strategies to find the optimal offers. At the end of this section, we discuss the computational complexity of this bargaining approach. At first, we calculate the reservation prices of the IP. To calculate the opportunity cost $v_n^{\mathrm{IP,opp}}$, we use backward induction. The backward induction consists of an initialization step and the subsequent induction steps.

1) Initialization: Using the backward induction, the initialization starts in the last bargaining round N with SP N. The sequential bargaining mechanism ends when the IP finishes negotiations with SP N. Consequently, the bargaining with SP N does not influence other negotiations. As a result, $v_N^{\mathrm{IP,opp}} = 0$ holds. Therefore, we can use the bargaining with SP N as our starting point for the backward induction. The IP calculates its expected payoff to calculate the expected profit when bargaining with SP N. The expected value of a random variable X is denoted by $\mathbb{E}\{X\}$. For the IP, the expected payoff is given by

$$\begin{aligned}\pi_N^{\mathrm{IP}}(k_N^{\mathrm{IP}}) &= \mathbb{E}\{P_N^{\mathrm{IP}}(k_N^{\mathrm{IP}})\} \\ &= \int_{k_N^{\mathrm{IP}}}^{\infty}\left(\frac{k_N^{\mathrm{IP}}+k_N^{\mathrm{SP}}}{2}-v_N^{\mathrm{IP}}\right)p^{\mathrm{SP}}(k_N^{\mathrm{SP}})\,dk_N^{\mathrm{SP}}\end{aligned} \tag{19}$$

when bargaining with SP N. To maximize its expected payoff $\pi_N^{\mathrm{IP}}(k_N^{\mathrm{IP}})$, the IP has to optimize its offer k_N^{IP} before the bargaining. This is difficult due to the fact that (19) is dependent on the SP's offer k_N^{SP}, which is unknown to the IP. Intuitively, if the SP is willing to pay a lot for the service placement, i.e., k_N^{SP} is high, it is advantageous for the IP to make a higher offer. Conversely, if the SP is willing to pay little for the service placement, i.e., k_N^{SP} is small, it is advantageous for the IP to make a lower offer.

This is an interdependent problem, where the IP's optimal strategy depends on the SP's strategy and vice versa. For this, there are infinite equilibria for the interdependent strategies [35]. For a subset of offer strategies, namely offer strategies for which the offers of IP and SP increase strictly monotonically with the reservation prices except for bound values, we can find a unique equilibrium [36]. This is known as the Linear Equilibrium Strategy (LES). In the following, we argue that the IP and the SPs choose their offers according to the LES. This allows us to determine the IP's optimal offer k_N^{IP}.

As the SP N is the last SP in the bargaining, we know that $v_N^{\mathrm{IP,opp}} = 0$ holds. This is because SP N is the last SP in the bargaining sequence. Therefore, the resources not given to SP N are not used. Consequently, the reservation price $\tilde{v}_N^{\mathrm{IP}} = v_N^{\mathrm{IP}}$ contains only the resource prices, and the opportunity cost is zero. Considering the limited resources, a service placement is only possible if the IP has enough resources to satisfy the demand of SP N, that is, $X_N^{\mathrm{IP}} \geq X_N^{\mathrm{SP}}$. The expected payoff of the IP is given as

$$\pi_N^{\mathrm{IP}}(X_N^{\mathrm{IP}}) = \begin{cases}\pi^{\mathrm{IP,LES}}(v_N^{\mathrm{IP}}) & \text{if } X_N^{\mathrm{IP}} \geq X_N^{\mathrm{SP}} \\ 0 & \text{if } X_N^{\mathrm{IP}} < X_N^{\mathrm{SP}}\end{cases} \tag{20}$$

which is a step function of the available resources X_N^{IP} in the last bargaining round N.

2) Backward Induction Steps: To employ the LES mechanism, all left to do is calculate the reservation prices $\tilde{v}_n^{\mathrm{IP}}$ of the bargaining round with SP n. We calculate this by backward induction. Instead of calculating v_n^{IP} as a scalar value for the bargaining with SP n, we determine the reservation price as a function of the available resources X_n^{IP}, i.e., $v_n^{\mathrm{IP}}(X_n^{\mathrm{IP}})$. The IP can determine its reservation price by looking up the value assigned to $X_{\mathrm{IP},n}$.

Consider now that the IP bargains with SP $N-1$. The IP has already concluded the bargainings with SP 1 to SP $N-2$. At the beginning of the negotiation, the IP has resources X_{N-1}^{IP} available. The outcome of the negotiation influences the resources the IP has available for bargaining with SP N. If the negotiation is successful, the service of SP $N-1$ is placed on the cloudlet and requires $X_{\mathrm{SP},N-1}$ resources. The IP has then $X_{N-1}^{\mathrm{IP}} - X_{N-1}^{\mathrm{SP}}$ resources available for the bargaining with SP N. If there is no agreement between IP and SP $N-1$, the IP has X_{N-1}^{IP} resources available for negotiation with SP N. The opportunity cost $v_N^{\mathrm{IP,opp}}$ describes the expected payoff the IP cannot realize due to a placement of SP $N-1$.

The reservation price of the IP in bargaining round $N-1$ is calculated as follows. First, we determine the opportunity cost

$$v_{N-1}^{\mathrm{IP,opp}} = \pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) - \pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}} - X_{N-1}^{\mathrm{SP}}), \tag{21}$$

which describes the expected payoff the IP cannot realize due to a placement of SP $N-1$. This is the difference between the expected payoff $\pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}})$ when no agreement is made, and the expected payoff $\pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}} - X_{N-1}^{\mathrm{SP}})$ when the service of SP $N-1$ is placed.

Now we can calculate the reservation price function

$$\begin{aligned}\tilde{v}_{N-1}^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) &= v_{N-1}^{\mathrm{IP}} + v_{N-1}^{\mathrm{IP,opp}} \\ &= v_{N-1}^{\mathrm{IP}} + \pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) \\ &\quad - \pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}} - X_{N-1}^{\mathrm{SP}}),\end{aligned} \tag{22}$$

which, according to (17), is the sum of the cost for the resources v_{N-1}^{IP} and the opportunity cost $v_{N-1}^{\mathrm{IP,opp}}$.

Based on the reservation price function, we can calculate the expected payoff function as

$$\begin{aligned}&\pi_{N-1}^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) \\ &= \begin{cases}\pi^{\mathrm{IP,LES}}(v_{N-1}^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}})) & \text{if } X_{N-1}^{\mathrm{IP}} \geq X_{N-1}^{\mathrm{SP}} \\ 0 & \text{if } X_{N-1}^{\mathrm{IP}} < X_{N-1}^{\mathrm{SP}}.\end{cases}\end{aligned} \tag{23}$$

The expected payoff function describes the additional payoff the IP can expect by bargaining with SP $N-1$ compared to a scenario where SP $N-1$ does not exist.

The aggregated expected payoff function is the payoff the IP expects from the remaining bargaining process. It is the sum of all individual expected payoff functions

$$\pi_{N-1}^{\mathrm{IP,agg}}(X_{N-1}^{\mathrm{IP}}) = \pi_{N-1}^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) + \pi_N^{\mathrm{IP}}(X_{N-1}^{\mathrm{IP}}) \tag{24}$$

with the available resources X_{N-1}^{IP} in bargaining round $N-1$.

Analogously, the backward induction is repeated until the bargaining round $n = 1$ is reached. After finishing the backward induction, the IP has calculated its reservation price functions $v_n^{\mathrm{IP}}(X_n^{\mathrm{IP}})$ for every bargaining round $n \in \{1, ..., N\}$.

3) IP's offer strategy: The last missing step for the IP is to calculate its optimal offer for the bargaining based on its reservation price functions $v_N^{\mathrm{IP}}(X_n^{\mathrm{IP}})$ from the backward induction. Using the calculated values of $v_N^{\mathrm{IP}}(X_n^{\mathrm{IP}})$ and the belief (12) about the SP's reservation price, we can determine the optimal offer k_n^{IP} in bargaining round n.

Theorem 1. *The optimal offer of the IP in round n according to the Linear Equilibrium Strategy is given by*

$$k_n^{\mathrm{IP}} = \begin{cases}\frac{2}{3}\underline{v}_n^{\mathrm{SP}} + \frac{1}{3}s_0 & \text{for } \tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}}) \leq \max(s_0, \underline{v}_n^{\mathrm{SP}}) \\ & \qquad -s_0 + \underline{v}_n^{\mathrm{IP}} \\ \frac{2}{3}(\tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}}) - \underline{v}_n^{\mathrm{IP}}) + s_0 & \text{else}\end{cases} \tag{25}$$

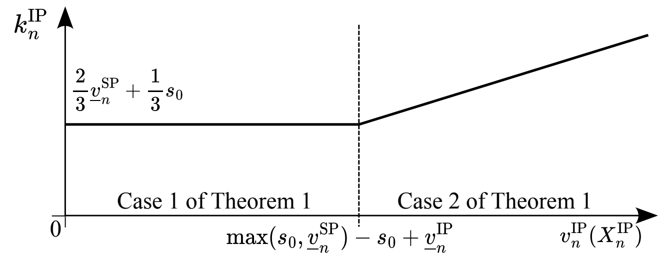


Fig. 2: The IP's optimal offer strategy according to Theorem 1 for a varying reservation price $\tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}})$ of the IP.

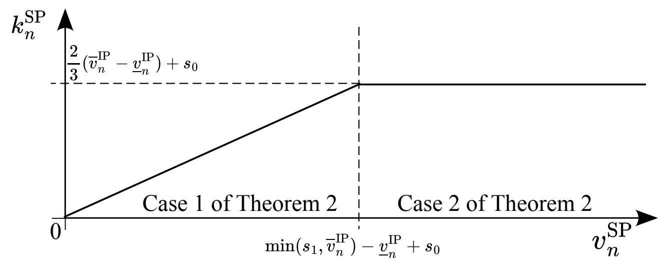


Fig. 3: The SP's optimal offer strategy according to Theorem 2 for a varying reservation price v_n^{SP} of the SP.

with $s_0 = \frac{3}{4}\underline{v}_n^{\mathrm{IP}} + \frac{1}{4}\overline{v}_n^{\mathrm{SP}}$.

Proof. The proof is analogous to the proof found in [36]. □

The optimal offer strategy of the IP is shown in Fig. 2. The optimal offer strategy is a piece-wise linear function of the IP's reservation price, which is common for offer strategies under IAs [36]. For the analysis of the IP's strategy, it is important to remember that a higher offer k_n^{IP} is better for the IP, as the payment from the SP to the IP is the mean value of the IP's and the SP's offers, as seen in (18).

The optimal offer k_n^{IP} from Theorem 1 consists of two cases: In the first case, the IP's reservation price is below a threshold $\tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}}) \leq \max(s_0, \underline{v}_n^{\mathrm{SP}}) - s_0 + \underline{v}_n^{\mathrm{IP}}$. This threshold depends on the SP's estimation of the lower bound $\underline{v}_n^{\mathrm{IP}}$ of the IP's reservation price. When this condition is met, the optimal offer is fixed at $k_n^{\mathrm{IP}} = \frac{2}{3}\underline{v}_n^{\mathrm{SP}} + \frac{1}{3}s_0$, independent of the IP's actual reservation price. This allows the IP to exploit the SP's underestimation of $\underline{v}_n^{\mathrm{IP}}$, making an offer higher than what would be expected based on the IP's true reservation price. In the second case, where the IP's reservation price exceeds the threshold $\tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}}) > \max(s_0, \underline{v}_n^{\mathrm{SP}}) - s_0 + \underline{v}_n^{\mathrm{IP}}$, the IP's offer increases linearly with the reservation price $\tilde{v}_n^{\mathrm{IP}}(X_n^{\mathrm{IP}})$. In this case, the optimal offer increases because of the IP's growing opportunity and placement costs.

C. The strategies of the service providers

The SP n has no information about the availability of the IP's resources X_n^{IP} in the current bargaining round n, i.e., the SP n does not know whether the computation resources are plenty or scarce. Furthermore, the SP n has no information about the costs associated with the resource usage of the IP. For the SP n, the expected payoff is given by

$$\begin{aligned}\pi^{\mathrm{SP}}(k_n^{\mathrm{SP}}) &= \mathbb{E}\{P_n^{\mathrm{SP}}(k_n)\} \\ &= \int_{-\infty}^{k_n^{\mathrm{SP}}} \left(v_n^{\mathrm{SP}} - \frac{k_n^{\mathrm{IP}} + k_n^{\mathrm{SP}}}{2}\right) p^{\mathrm{IP}}(k_n^{\mathrm{IP}}) dk_n^{\mathrm{IP}}\end{aligned} \tag{26}$$

which is the expected value of the payoff based on its offer k_n^{SP}. Based on the belief (11) about the IP's reservation price and the reservation price v_n^{SP} of SP n, the SP n can calculate its optimal offer.

Theorem 2. *The optimal offer of the SP n according to the Linear Equilibrium Strategy is given by*

$$k_n^{\mathrm{SP}} = \begin{cases} \frac{2}{3}v_n^{\mathrm{SP}} + \frac{1}{3}s_0 & \text{for} \quad v_n^{\mathrm{SP}} \leq \min(s_1, \overline{v}_n^{\mathrm{IP}}) \\ & \qquad\qquad -\underline{v}_n^{\mathrm{IP}} + s_0 \\ \frac{2}{3}(\overline{v}_n^{\mathrm{IP}} - \underline{v}_n^{\mathrm{IP}}) + s_0 & \text{else} \end{cases} \tag{27}$$

with $s_0 = \frac{3}{4}\underline{v}_n^{\mathrm{IP}} + \frac{1}{4}\overline{v}_n^{\mathrm{SP}}$ *and* $s_1 = \frac{3}{4}\overline{v}_n^{\mathrm{SP}} + \frac{1}{4}\underline{v}_n^{\mathrm{IP}}$.

Proof. The proof is analogous to the proof found in [36]. □

The optimal offer strategy of the SP is shown in Fig. 3. For the analysis of the SP's strategy, it is important to remember that a lower offer k_n^{IP} is favorable for the SP, as a lower k_n^{IP} results in a lower payment from the SP to the IP according to (18). In Fig. 3, we can see also a piece-wise linear optimal strategy. For lower reservation prices of the SP, the optimal offer increases linearly with the SP's reservation price v_n^{SP}. After the threshold $v_n^{\mathrm{SP}} \geq \min(s_1, \overline{v}_n^{\mathrm{IP}}) - \underline{v}_n^{\mathrm{IP}} + s_0$, the SP does not increase its offer any further and the SP's optimal offer is fixed to $k_n^{\mathrm{SP}} = \frac{2}{3}(\overline{v}_n^{\mathrm{IP}} - \underline{v}_n^{\mathrm{IP}}) + s_0$. This is an effect of the IAs, as the SP can exploit the limited information of the IP.

D. Computational complexity

To execute the LES bargaining mechanism, we need to calculate the reservation price functions and the IP's expected payoff functions. Each of these functions from (23) is a step function as every function is the sum and concatenation of other step functions. To fully describe such a function, each resource amount is stored where the function changes its value, and the corresponding function value is sufficient. If the SPs can demand an arbitrary amount of resources $X_n^{\mathrm{SP}} \in \mathcal{R}$, the number of resource amounts where the reservation price functions change value grows exponentially with each step of the backward induction. Thus, the bargaining mechanism has a computational complexity of $\mathcal{O}(2^N)$.

We argue that, in reality, it is reasonable to assume that resources are organized in discrete computation instances, like virtual machines at a cloudlet. This is common practice for MEC providers [38]. By making this assumption, the possible combinations of demands X_n^{SP} of the SPs and the IP's available resources X^{IP} reduce to an integer number of these computing instances. Therefore, we can significantly reduce the computational complexity of our proposed NBI-LES approach. In our case, the SPs can only demand multiples of a single computational instance, i.e., only integers of instances. As a result, the step functions of the IP's reservation price can only change their value at integer resource amounts. This reduces the computational complexity to $\mathcal{O}(X_{\mathrm{IP}} \cdot N)$ and allows for computation of the bargaining mechanism when the number N of SPs is large.

E. Effects of the bargaining order

The order of SPs in the sequential bargaining process can influence the outcome at step n due to the IP's opportunity cost $v_n^{\mathrm{IP,opp}}$, which depends on n. This can lead to variations in the utility for both the SP and the IP during individual negotiations. In the complete information case discussed in [41], an early-mover advantage is observed, where the first SP enjoys a higher expected payoff due to an increased likelihood of reaching an agreement and securing service placement.

However, our primary focus in this work is on evaluating the efficiency and fairness of the overall network. Therefore, we concentrate on aggregate metrics such as total utility, resource utilization, and average resource prices. As shown in [41], the bargaining order does not affect these aggregate metrics, nor does it impact the overall efficiency of the proposed solution.

IV. Experimental evaluation

In this section, we evaluate the performance of the proposed NBI-LES approach and compare it to state-of-the-art algorithms using a real service usage dataset. We will first discuss the service usage dataset and the selection of the different MEC service use cases. Afterward, we present the model for the cost factors, IAs, and resources.

A. Case Study and Data Set

We obtained data on the use of services from different SPs to study the performance and effectiveness of our approach in the simulation. We are unaware of a public dataset that provides usage information at the appropriate granularity for various SPs with heterogeneous use cases. We collected the data in a field study to use reasonably realistic smartphone usage as a basis for the simulations. Overall, $K = 544$ participants from the US took part in the study for about three weeks, starting on 30th of November until 23rd of December 2022. We aggregated the data over all $K = 544$ users and 36 services to obtain an average service usage time per service.

We consulted related literature [5], [6], [48]–[50] and [51] to gain an overview of possible use cases for the MEC scenario. We chose a subset of use cases for simplicity and analyzed them regarding their traffic and computational load. We determine service use case categories according to computational and data traffic demands, which can be either high or low. Table III visualizes three of these combinations providing possible use cases with different requirements regarding computation and traffic, i.e., computation high and traffic low (computation-heavy), computation low and traffic high (traffic-heavy), and computation high and traffic high (multiple-heavy). We excluded the fourth imaginable use case with low computation and traffic load, as we consider this use case least appropriate for MEC as there is only low potential for efficient use in MEC. Hence, only three use case categories remain: computation-heavy, traffic-heavy, and multiple-heavy. We selected the $N = 16$ most popular services in our dataset from these use case categories. We chose the use cases "mobile AR", "video streaming", and "video analytics". Based on the usage time duration of a service, we can determine the computational resource demand X_n^{SP} and data traffic of each service type. The type of service and the associated MEC use case plays a particular role, as the resource demand X_n^{SP} differs between different service types.

Regarding the computation and traffic reduction by offloading a service to the edge in respective use cases, we based our parameters on existing studies [5], [6], [48], [49] and [51], that have implemented and evaluated similar services. When the studies above presented different values for the computational or data traffic demands, we averaged these values for the respective use cases. We measure the computational demand X_n^{SP} in virtual CPUs (vCPUs). First, we consider the use case "mobile AR" from the use case category "computation-heavy". In case of a mobile AR game, the data traffic demand is as low as $0.028\,\mathrm{MB/h}$ [6], while a Full HD video stream might require $4.6\,\mathrm{GB/h}$, e.g. when a video stream is augmented live. We average this to a data traffic demand of $D_n^{\mathrm{SP}} = 2.3\,\mathrm{GB/h}$. As the "mobile AR" use case requires large amounts of computation resources, we assume $X_n^{\mathrm{SP}} = 4$ vCPUs per service instance. For the use case "video streaming" from the use case category "traffic-heavy", we assume $X_n^{\mathrm{SP}} = 1$ vCPU per service for a relatively low computation load. At the same time, we consider data traffic demand to be $4.6\,\mathrm{GB/h}$, assuming Full HD video streaming. Finally, we inspect the use case "video analytics" from the use case category "multiple-heavy". For this use case, we estimate a 4K video stream to require between $4.6\,\mathrm{GB/h}$ and $15\,\mathrm{GB/h}$, averaging to about $D_n^{\mathrm{SP}} = 9.65\,\mathrm{GB/h}$ of data traffic demand. Simultaneously, we assume this to be a computation-heavy use case. Hence, we presume $X_n^{\mathrm{SP}} = 4$ vCPUs per service instance.

We estimate the cost factors based on real-world IPs like [38] and [39]. We differentiate cost factors associated with cloud computing $\alpha_n^{\mathrm{SP}}, \beta_n^{\mathrm{SP}}, \Phi_n^{\mathrm{SP}}$ and the cost factors $\alpha^{\mathrm{IP}}, \Phi_n^{\mathrm{IP}}$ associated with MEC, which are presented in Table IV.

B. Evaluation Setup

In our evaluation, we consider the service placement to be performed every hour for a $T = 24$ days, resulting in a total of $T \cdot 24 = 576$ different service placements. We assume a constant number X^{IP} of available vCPUs at the cloudlet for the entire period. The order of SPs in each bargaining procedure is randomly selected to mitigate the effects of bargaining order, as discussed in Sec. III-E. For a consistent comparison, the same bargaining order is used for all approaches considered.

We determine the resource demands X_n^{SP} of any given SP through two variables. The first is the service instance requirement (SIR), which indicates how many computation resources must be allocated to the service-specific software

TABLE III: Overview of the analyzed use cases.

Use Case	Domain	Computation	Traffic	Traffic reduction	CPUs per Service	Traffic per Service
computation-heavy	Mobile AR	High	Low	0.9 [5]	4vCPU	$2.3\,\mathrm{GB/h}$
traffic-heavy	Video Streaming	Low	High	0.3 [48]	1vCPU	$4.5\,\mathrm{GB/h}$
multiple-heavy	Video Analytics	High	High	0.8 [49]	4vCPU	$9.65\,\mathrm{GB/h}$

TABLE IV: Evaluation parameters

Symbol	Description	Value	Source
T	Considered time period	$24\,\mathrm{d}$	
t	Bargaining interval	$1\,\mathrm{h}$	
N	Number of SPs	16	
σ^{SP}	Uncertainty about the SPs' reservation prices v_n^{SP}	0.5	
σ^{IP}	Uncertainty about the IP's reservation prices v_n^{IP}	0.5	
α_n^{SP}	Cost per computation resource at the cloud	0.056 USD per hour	[38]
β_n^{SP}	Cost for communication between user and the cloud	0.09 USD per GB	[38]
Φ_n^{SP}	Cost for storage and maintenance of the service provided by SP n	0.000208 USD/GB per hour	[39]
Φ_n^{IP}	Infrastructure cost for storage and maintenance	0.000208 USD/GB per hour	[39]
α^{IP}	Cost per computation resource at the cloudlet	0.0416 USD per hour	[38]

for each category. For example, one service instance of the use case "mobile AR" may require 4 vCPUs, whereas the use case "video streaming" requires only 1 vCPU per service instance. The second important variable is the total service usage duration t^{service}, which is the total time the users access a service in a given hour. We assume that a duration of $t^{\mathrm{service}} = 3600\,\mathrm{s}$ of app usage requires one service instance to process the user demands. We then calculate the total demand as $X_n^{\mathrm{SP}} = \lceil \mathrm{SIR} \cdot \frac{t^{\mathrm{service}}}{3600} \rceil$. Since the IP sells vCPUs only in discrete units, we round up the demand to the next highest integer to ensure sufficient computation resources are allocated to each service instance.

To determine the reservation prices of the SPs, we first calculate the reservation price v_n^{SP} of each SP according to (4). Additionally, we assume that noise is present to model the uncertainty of the reservation prices. We assume that to each reservation price v_n^{SP}, a uniformly distributed random variable $v_n^{\mathrm{Noise}} \sim \mathcal{U}(-\sigma^{\mathrm{SP}} v_n^{\mathrm{SP}}, \sigma^{\mathrm{SP}} v_n^{\mathrm{SP}})$ is added. If not stated otherwise, we set $\sigma^{\mathrm{SP}} = 0.5$, i.e., the reservation price fluctuates $50\,\%$ around the reservation price v_n^{SP}.

For the IP, we calculate the physical cost v_n^{IP} for service placement according to (8). We assume that each SP can accurately estimate this cost because each SP knows its own computation, communication, and storage resource demand and can attain the physical computation cost per resource through other edge computation providers. In a scenario with unlimited resources, the physical cost v_n^{IP} corresponds to the reservation price $\tilde{v}_n^{\mathrm{IP}}$ of the IP, i.e., $\tilde{v}_n^{\mathrm{IP}} = v_n^{\mathrm{IP}}$, as the SPs do not need to compete for the resources in the case of unlimited resources. Therefore, the physical cost v_n^{IP} of computation is the lower bound of the IP's reservation price. The other extreme scenario describes the case that resources are very scarce, i.e., the sum of the SPs' resource demands $\sum_{n=1}^{N} X_n^{\mathrm{SP}}$ is much larger than the number X^{IP} of available vCPUs. In such a case, the highest price the IP could reasonably demand is the upper bound of the SP's reservation price v_n^{SP}. The upper bound of the IP's reservation price equals, therefore, the SP's reservation price upper bound. Consequently, each SP can model its belief function of (11) as a uniform distribution between v_n^{IP} and v_n^{SP}.

C. Evaluation metrics

The IP and the SPs have different technical and economic parameters, so assessing the system's performance depends on the considered perspective. We argue that different evaluation metrics must be considered to measure the system's performance.

1) *Resource utilization:* The resource utilization $R(\boldsymbol{x})$ is defined as the number of computation resources, i.e., vCPUs, used in the service placement. This is defined as $R(\boldsymbol{x}) = \sum_{n=1}^{N} x_n X_n^{\mathrm{SP}}$.
2) *Cloudlet service time:* We consider the time users access services placed on the cloudlet. A higher cloudlet service time corresponds to a higher service quality for the users, as more service requests are processed directly on the cloudlet.
3) *Reduced data traffic to the cloud:* We consider the amount of data that can be processed locally on the cloudlet instead of sent to the central cloud.
4) *Social Welfare:* As described in Section II-E, social welfare is used to measure the system's efficiency.
5) *Jain's index:* As described in Section II-F, we use Jain's index to measure the system's fairness.
6) *Price of the resources:* We consider the average price k_n per resource.
7) *Profits of the IP and SPs:* We consider the individual payoffs generated by the service placement of the SPs, see (3), and the IP, see (7).

D. Baseline Algorithms

To evaluate our proposed NBI-LES, we compare it with traditional pricing approaches. We use the following algorithms to benchmark our proposed NBI-LES.

- *Fixed price per resource determined by the IP*, which is abbreviated as *Fixed Price*: The IP sets a fixed price p per vCPU for every SP in this method. Each SP then decides if it accepts a service placement for the given price. This approach is derived from state-of-the-art approaches [23] and [25] and is the most commonly used in practice by major cloud providers like AWS [39] or Azure. An SP n will agree to service placement if the payment $k_n = pX_n^{\mathrm{SP}}$ for service placement is smaller

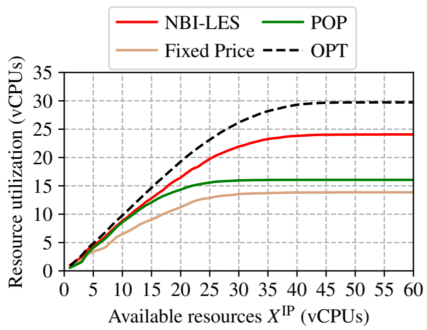


Fig. 4: Average resource utilization in vCPUs as a function of X^{IP}.

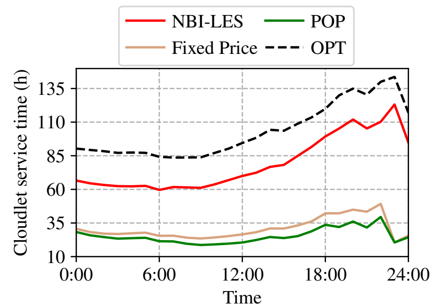


Fig. 5: Average cloudlet service time as a function of the time of day, $X^{\mathrm{IP}} = 30$.

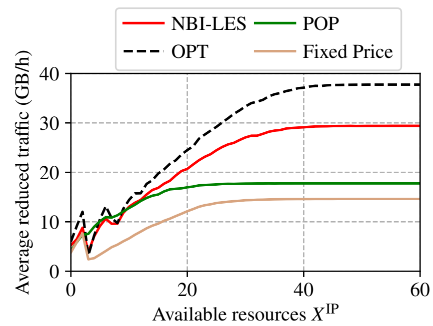


Fig. 6: Reduced data traffic to the cloud in $\mathrm{GB/h}$ as a function of X^{IP}.

than its reservation price, i.e., $v_n^{\mathrm{SP}} > k_n$. For better comparison, we assume that the SPs decide on the service placement in the same order they bargain with the IP in our LES mechanism. The SPs' and IP payoffs are calculated according to (3) and (7). The IP will choose the price p per vCPU to maximize its expected payoff for the service placement process considering the IAs. This price p depends on the number X^{IP} of available resources and is calculated through backward induction.

- *Centralized popularity-aware service placement*, which is abbreviated as *POP*. Using this approach, the services with the highest usage time t^{service} are placed on the cloudlet by a centralized network orchestrator [52] with full information available. The IP charges for each vCPU a fixed price $p = 2v_n^{\mathrm{IP}}$, which reflects the IP's reservation price and a profit margin. The payment for the service placement is then given by $k_n = pX_n^{\mathrm{SP}}$. This approach only considers the popularity of each service, i.e., it aims to maximize the probability that a service request is processed on the cloudlet. However, it neglects the aspects of efficient resource utilization and disregards the economic considerations of the SPs and the IP.
- *Centralized optimal service placement*, which is abbreviated as *OPT*: The OPT strategy describes which service placement decisions $\boldsymbol{x}$ a centralized network orchestrator with perfect information, i.e., without IAs, would take to maximize the social welfare $\Psi(\boldsymbol{x})$ (13). This is a state-of-the-art approach from [12], which we adapted to the given scenario. Resource demands X_n^{SP} and reservation prices $v_n^{\mathrm{SP}}, v_n^{\mathrm{IP}}$ are known to a central network orchestrator. The payments for the service placements are chosen to be $k_n = \frac{v_n^{\mathrm{SP}} + v_n^{\mathrm{IP}}}{2}$, i.e., the profit is split equally between the IP and the SP. We can formulate the centralized service placement problem as

$$\max_{\boldsymbol{x}} \quad \Psi(\boldsymbol{x}) \tag{28}$$

$$\text{s. t.} \quad x_n \in \{0, 1\} \tag{29}$$

$$\sum_{n=1}^{N} x_n \cdot X_n^{\mathrm{SP}} \leq X^{\mathrm{IP}}, \tag{30}$$

where $\Psi(\boldsymbol{x})$ is the social welfare from (13). The resulting centralized optimal service placement $\boldsymbol{x}$ is obtained using a solver.

E. Results and Discussion

In Fig. 4, we assess the resource utilization for the proposed NBI-LES, Fixed Price, POP, and OPT algorithm for a varying number X^{IP} of available vCPUs. For a low number of available vCPUs, i.e., $X^{\mathrm{IP}} < 10$, the NBI-LES, POP, and OPT algorithms achieve a high resource utilization. For a high number of resources, the proposed NBI-LES algorithm achieves a resource utilization within $83\,\%$ of the OPT algorithm and a $50\,\%$ ($71\,\%$) higher resource utilization than the POP and Fixed Price algorithm respectively. The POP and OPT outperform the Fixed Price algorithm as they require a central network orchestrator and the SPs and the IP to report all their information. The resource utilization is not equal to $100\,\%$ even for the OPT algorithm, as the service placement is not economically viable for all SPs for every hour. For some services, where either the current demand is low or the benefits from MEC are not large, it is advantageous to use cloud resources instead of service placement in MEC.

The cloudlet service time over the course of a day for $X^{\mathrm{IP}} = 30\,$vCPUs is shown in Fig. 5. For this, the data over the whole period of 24 days has been averaged for each hour of the day. The maximum service time of the cloudlet is at around 23:00 hours. For the OPT approach, the cloudlet service time at 23:00 is 137 hours, i.e., 137 hours of service usage time from all users, and for all N SPs between 23:00 and 0:00 has been performed on this cloudlet. The NBI-LES has a cloudlet service time of 123 hours, $89.8\,\%$ of the OPT algorithm. The Fixed Price and POP algorithms perform significantly worse and achieve only $40.7\,\%$ ($30\,\%$) of the cloudlet service time the proposed NBI-LES provides. Both the Fixed Price and the POP algorithms suffer because the price k_n for the computation resources is not adjusted for each SP. This makes service placement beneficial for SPs with a high reservation price v_n^{SP}, which is above the price. The price from the Fixed Price algorithm is calculated according to the current total demand for resources. The price of the POP algorithm only depends on the profit margin and physical cost of the IP.

We show the reduced data traffic to the cloud in Fig. 6. The reduced data traffic fluctuates for a low number of vCPUs, $X^{\mathrm{IP}} < 10$. This is because several computation-heavy services

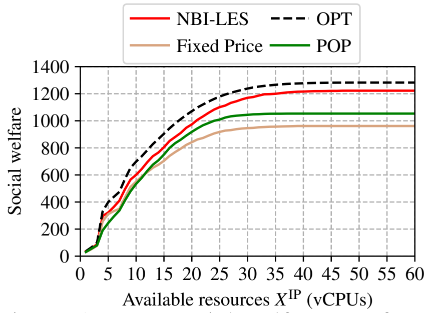


Fig. 7: Average social welfare as a function of X^{IP}.

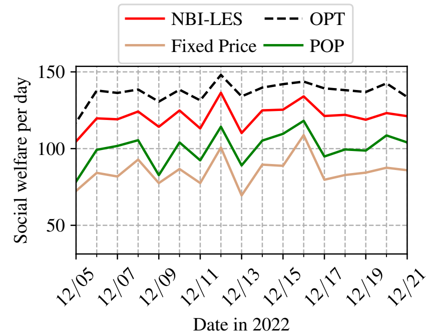


Fig. 8: Average social welfare as a function of the day, $X^{\text{IP}} = 30$.

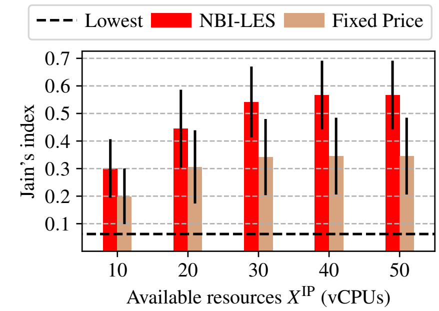


Fig. 9: Average of the Jain's index as a function of X^{IP}, the black bars indicate the variance.

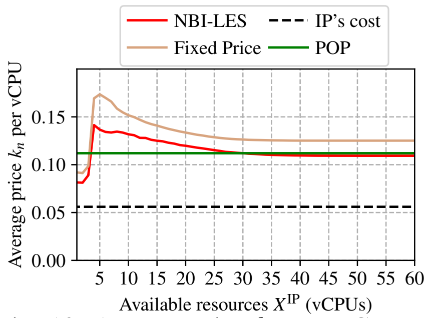


Fig. 10: Average price k_n per vCPU as a function of the available vCPUs.

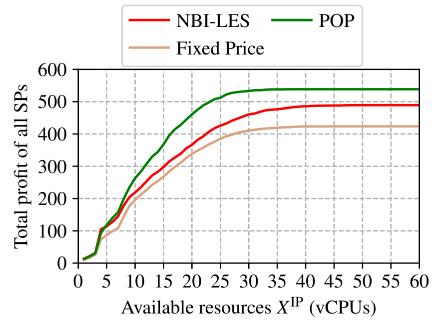


Fig. 11: Profit of the SP as a function of the available vCPUs.

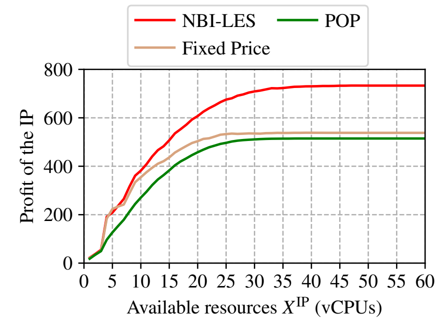


Fig. 12: Profit of the IP as a function of the available vCPUs.

can only be placed if the available vCPUs exceed the demand. Therefore, if the number of vCPUs is increased, a service with higher computational demand will replace a service with higher data traffic demands, decreasing the reduced data traffic to the cloud. For a sufficient number $X^{\text{IP}} > 40$ of vCPUs, the OPT algorithm saves $38\,\text{GB/h}$. The NBI-LES achieves $79\,\%$ of data reduction compared to the OPT approach and performs $60\,\%$ ($100\,\%$) better than the POP and Fixed Price algorithm, respectively.

We assess the social welfare of (13) for the proposed NBI-LES, Fixed Price, POP, and OPT algorithm in Fig. 7 for a varying number of available computation instances X^{IP}. The social welfare is monotonically increasing since more available computation resources X^{IP} enable more service placements. The higher number of service placements corresponds with an increase in payoff for both IPs and SPs, which in turn increases social welfare. When resources are scarce, SPs willing to pay a higher price, i.e., with a higher reservation price v_n^{SP}, will receive the service placements. Furthermore, the social welfare approaches a limit when available resources X^{IP} increase. This limit exists due to the finite demand X_n^{SP} of SPs. We can see that our proposed NBI-LES algorithm performs $14.2\,\%$ better than the Fixed Price approach, $26.3\,\%$ better than the POP algorithm, and achieves $96\,\%$ of the optimal social welfare. From these results, we can also conclude that a uniform price for all SPs, as used by the Fixed Price and POP approach, leads to a decrease of social welfare.

In Fig. 8, the social welfare is shown over the whole period of T for the NBI-LES, Fixed Price, POP, and OPT approach. Social welfare varies daily as the users' service usage and, therefore, the SPs' resource demands change. The proposed NBI-LES algorithm provides higher social welfare than the POP and Fixed Price algorithms on days with high demand as well as on days with low demand.

Fig. 9 shows the Jain's index (14) as a function of the available vCPUs for the proposed NBI-LES algorithm and the Fixed Price algorithm. As the OPT and the POP algorithms are centralized approaches, we do not consider their fairness. The black dashed line indicates the least fair solution, i.e., $\mathcal{J}(\boldsymbol{x}, \boldsymbol{k}) = \frac{1}{N}$, where only one stakeholder takes all the profit. For a lower number of resources $X^{\text{IP}} = 10$, the proposed NBI-LES algorithm achieves an average Jain's index of 0.3, while the Fixed Price only achieves 0.2. These low numbers are because only a few services can be placed on the cloudlet, which results in an unfair distribution that benefits the SPs with the highest profit from service placement. For more resources, the fairness increases, as more SPs can use MEC and place their services on the cloudlets. Our proposed NBI-LES algorithm significantly increases the fairness compared to the Fixed Price approach because the IP adapts its price for each SP. This is beneficial for the IP as well as the SPs because more service placements are possible. The fairness measured by the Jain's index does not approach perfect fairness $\mathcal{J}(\boldsymbol{x}, \boldsymbol{k}) = 1$ due to the IAs.

Fig. 10 shows the average price k_n paid per vCPU. For comparison, the blacked dashed line shows the IP's physical cost v_n^{IP} (8) per computation resource. Both the Fixed-Price Approach and the proposed NBI-LES approach adapt their

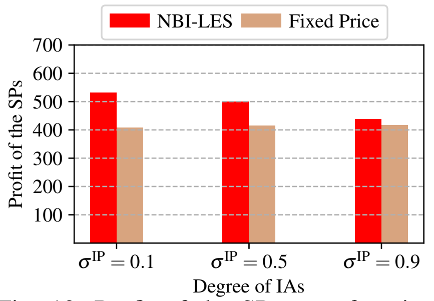


Fig. 13: Profit of the SPs as a function of the available information σ^{IP}.

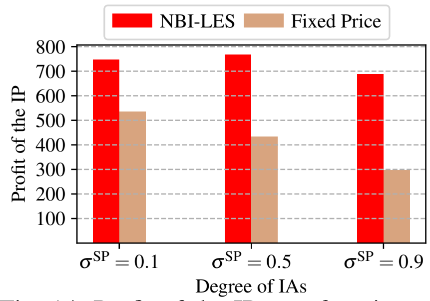


Fig. 14: Profit of the IP as a function of the available information σ^{SP}.

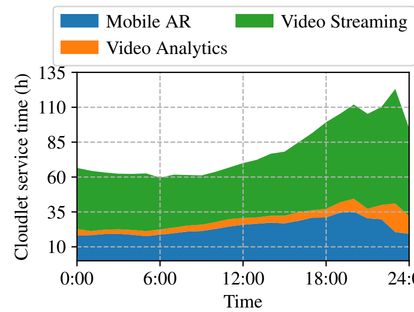


Fig. 15: Cloudlet service time for the proposed NBI-LES algorithm as a function of the time of day.

prices to the current demand and available vCPUs. For a lower amount of available vCPUs, the price is higher. The price is lower for $X^{\mathrm{IP}} < 4$, as only SPs with a very low computational demand can place the service on the cloudlet. The price of the proposed NBI-LES algorithm is lower than that of the Fixed Price approach, which enables SPs with a lower reservation price to achieve a service placement. This increases the fairness of the service placement and allows for better resource utilization. The POP algorithm's fixed price is $p = 2v_n^{\mathrm{IP}}$, corresponding to the average price per vCPU of the proposed NBI-LES algorithm.

Fig. 11 depicts the sum of the profit of all SPs for the NBI-LES, POP, and Fixed Price algorithm. The POP algorithm provides a larger profit for the SPs, as the prices are lower when the resources are scarce, as shown in Fig. 10. Furthermore, the price is not adapted for each SP; therefore, SPs with a high reservation price profit from the low price that the POP algorithm provides. As the POP algorithm is a centralized approach, no IAs between the SPs and the IP are considered, making real-world implementation difficult. The proposed NBI-LES algorithm outperforms the Fixed Price approach by $16.6\,\%$ regarding the SPs' profit.

Fig. 12 depicts the profit of the IP from the service placement for the entire time period T depending on the number of available vCPUs. When the IP has more vCPUs available at the cloudlet, the NBI-LES algorithm generates higher profit for the IP. This leads to higher price offers k_n^{IP} of the IP for SPs with higher expected reservation prices and lower price offers for SPs with lower expected reservation prices. In comparison, when the IP sets a fixed price for all SPs, it must choose between price per resource and the number of service placements. If the IP sets a high price, it will receive high payments, but from only a few SPs, thus not using many cloudlet resources. In contrast, if the IP sets a lower price, more services will be placed on the cloudlet, but the profit per resource is lower. This lower flexibility of the Fixed Price method leads to lower overall profits for the IP for the Fixed Price approach. In the comparison between the IP's profit from Fig. 12 and the SPs' profit from Fig. 11, we can see that the SPs have a lower total profit than the IP. This can be explained by the fact that the SPs are unaware of the number X^{IP} of available vCPUs, which is a disadvantage in the bargaining procedure.

Furthermore, we investigate the effect of the degree of IAs on the performance of the NBI-LES and Fixed Price approach. The centralized approaches have complete information. Therefore, the OPT and POP algorithms are not considered. We vary the information available σ^{IP} and σ^{SP} of the bargaining partner's reservation prices, where a larger value of σ^{IP} (σ^{SP}) corresponds to greater IAs. In the first experiment, shown in Fig. 13, we vary the uncertainty σ^{IP} about the IP's reservation price and analyze how this affects the profit of the SPs. We can see that for the NBI-LES approach, the profit of the SPs decreases with increasing σ^{IP}. For the Fixed Price approach, the SPs' profit is constant over σ^{IP}, as the SPs are not involved in the pricing of the resources. For the Fixed Price approach, the IP determines the price without involving the SPs.

In the second experiment, shown in Fig. 14, we vary the uncertainty σ^{SP} about the SPs reservation prices and analyze how this affects the profit of the IP. Here, we can see that the profit of the IP decreases much faster for the Fixed Price approach than for the NBI-LES with increasing IAs. The proposed NBI-LES is, therefore, more robust against IAs, as both the IP and the SPs are involved in the pricing, whereas in the Fixed-Price approach, the IP sets the price based on its own information. If the information about the SPs' reservation prices is inaccurate, the Fixed Price approach performs significantly worse than the NBI-LES approach. In the proposed NBI-LES approach, the SPs *signal* their willingness to pay in the form of the offer k_n^{SP}, which reduces the IAs.

Finally, we analyze the average distribution of the cloudlet service time for the different use cases in Fig. 15 over the course of a day. We can see that the use case "video streaming" has approximately $60\,\%$ of the total cloudlet service time. This is because large amounts of video data can be stored directly at the edge, which is economically very efficient. Furthermore, we can see that different use cases have different usage patterns over the day. The maximum service usage for "mobile AR" is around 19:00, whereas for "video streaming" it is around 23:00. This highlights the importance of a dynamic service placement strategy, like the proposed NBI-LES algorithm, that adapts to users' current demands.

V. Conclusion and Future Work

We studied the MEC service placement problem in this paper, considering multiple stakeholders. We have analyzed each relevant stakeholder's technical and economic perspectives, namely the IPs and the SPs. Additionally, we have modeled IAs concerning the cost factors and the available resources of the respective bargaining partner. To solve the service placement problem, we argue that a fair and efficient solution is required. Fairness is required to ensure that all stakeholders profit from the service placement and, therefore, are motivated to participate. Efficiency is required to utilize the MEC network's resources to achieve the best performance in terms of cost reduction and data traffic reduction. We have proposed a novel bargaining approach, termed NBI-LES, which combines backward induction and Nash bargaining. We derive the LES, which represents the equilibrium for all the stakeholders in the bargaining procedure. Simulation results show that, compared to a traditional fixed price approach, our proposed NBI-LES algorithm performs significantly better regarding resource utilization, and the amount of data sent to the cloud can be significantly reduced.

Future works could analyze the impact of SPs forming groups or coalitions to improve their bargaining power in relation to the IP.

References

[1] H. Tabatabaee Malazi, S. R. Chaudhry, A. Kazmi, A. Palade, C. Cabrera, G. White, and S. Clarke, "Dynamic Service Placement in Multi-Access Edge Computing: A Systematic Literature Review," *IEEE Access*, vol. 10, pp. 32 639–32 688, Mar. 2022.

[2] B. Sonkoly, J. Czentye, M. Szalay, B. Németh, and L. Toka, "Survey on Placement Methods in the Edge and Beyond," *IEEE Commun. Surveys & Tutorials*, vol. 23, no. 4, pp. 2590–2629, Jul. 2021.

[3] B. Ali, M. A. Gregory, and S. Li, "Multi-Access Edge Computing Architecture, Data Security and Privacy: A Review," *IEEE Access*, vol. 9, pp. 18 706–18 721, Jan. 2021.

[4] S. Shi, V. Gupta, M. Hwang, and R. Jana, "Mobile VR on Edge Cloud: A Latency-Driven Design," in *Proc. of the 10th ACM Multimedia Syst. Conf. (MMSys)*, Amherst, Massachusetts, 2019, p. 222–231.

[5] N. Wang, B. Varghese, M. Matthaiou, and D. S. Nikolopoulos, "ENORM: A framework for edge node resource management," *IEEE Trans. on Services Comput.*, vol. 13, no. 6, pp. 1086–1099, Nov. 2020.

[6] A. Sterz, P. Felka, B. Simon, S. Klos, A. Klein, O. Hinz, and B. Freisleben, "Multi-Stakeholder Service Placement via Iterative Bargaining With Incomplete Information," *IEEE/ACM Trans. on Netw.*, vol. 30, no. 4, pp. 1822–1837, Mar. 2022.

[7] B. Simon, S. Dongare, T. Mahn, A. Ortiz, and A. Klein, "Delay- and Incentive-Aware Crowdsensing: A Stable Matching Approach for Coverage Maximization," in *Proc. of IEEE Int. Conf. on Commun. (ICC)*. IEEE, 2022, pp. 2984–2989.

[8] G. Qu, Z. Lin, F. Liu, X. Chen, and K. Huang, "TrimCaching: Parameter-Sharing AI Model Caching in Wireless Edge Networks," in *Proc. of IEEE 44th Int. Conf. on Distributed Computing Systems (ICDCS)*, 2024, pp. 36–46.

[9] J. Wu, L. Wang, Q. Pei, X. Cui, F. Liu, and T. Yang, "HiTDL: High-Throughput Deep Learning Inference at the Hybrid Mobile Edge," *IEEE Trans. on Parallel and Distributed Systems*, vol. 33, no. 12, pp. 4499–4514, Dec. 2022.

[10] J. Gedeon, F. Brandherm, R. Egert, T. Grube, and M. Mühlhäuser, "What the Fog? Edge Computing Revisited: Promises, Applications and Future Challenges," *IEEE Access*, vol. 7, pp. 152 847–152 878, Oct. 2019.

[11] M. Satyanarayanan, "The Emergence of Edge Computing," *Computer*, vol. 50, no. 1, pp. 30–39, Jan. 2017.

[12] M. Chen, Y. Hao, L. Hu, M. S. Hossain, and A. Ghoneim, "Edge-CoCaCo: Toward Joint Optimization of Computation, Caching, and Communication on Edge Cloud," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 21–27, Jun. 2018.

[13] B. Gao, Z. Zhou, F. Liu, F. Xu, and B. Li, "An Online Framework for Joint Network Selection and Service Placement in Mobile Edge Computing," *IEEE Transactions on Mobile Computing*, vol. 21, no. 11, pp. 3836–3851, Mar. 2022.

[14] S. Deng, Z. Xiang, J. Taheri, M. A. Khoshkholghi, J. Yin, A. Y. Zomaya, and S. Dustdar, "Optimal Application Deployment in Resource Constrained Distributed Edges," *IEEE Trans. on Mobile Comput.*, vol. 20, no. 5, pp. 1907–1923, 2021.

[15] P. Jin, X. Fei, Q. Zhang, F. Liu, and B. Li, "Latency-aware VNF Chain Deployment with Efficient Resource Reuse at Network Edge," in *Proc. of IEEE Conf. on Comp. Commun. (INFOCOM)*, Jul. 2020, pp. 267–276.

[16] M. Li, Q. Zhang, and F. Liu, "Finedge: A Dynamic Cost-Efficient Edge Resource Management Platform for NFV Network," in *Proc. of IEEE/ACM 28th Int. Symposium on Quality of Service (IWQoS)*, Jun. 2020, pp. 1–10.

[17] X. Zhang, Z. Li, C. Lai, and J. Zhang, "Joint Edge Server Placement and Service Placement in Mobile-Edge Computing," *IEEE Internet of Things J.*, vol. 9, no. 13, pp. 11 261–11 274, Nov. 2022.

[18] Y.-Y. Shih, C.-Y. Wang, and A.-C. Pang, "Fog Computing Service Provision Using Bargaining Solutions," *IEEE Trans. on Services Comput.*, vol. 14, no. 6, pp. 1765–1780, Mar. 2021.

[19] F. Zafari, P. Basu, K. K. Leung, J. Li, D. Towsley, and A. Swami, "Resource Sharing in the Edge: A Distributed Bargaining-Theoretic Approach," *IEEE Trans. on Netw. and Service Management*, vol. 20, no. 4, pp. 4369–4382, Apr. 2023.

[20] T. Nisha, D. T. Nguyen, and V. K. Bhargava, "A Bilevel Programming Framework for Joint Edge Resource Management and Pricing," *IEEE Internet of Things J.*, vol. 9, no. 18, pp. 17 280–17 291, Feb. 2022.

[21] S. Chen, L. Jiao, F. Liu, and L. Wang, "EdgeDR: An Online Mechanism Design for Demand Response in Edge Clouds," *IEEE Trans. on Parallel and Distributed Systems*, vol. 33, no. 2, pp. 343–358, 2022.

[22] D. T. Nguyen, L. B. Le, and V. Bhargava, "Price-Based Resource Allocation for Edge Computing: A Market Equilibrium Approach," *IEEE Trans. on Cloud Comput.*, vol. 9, no. 1, pp. 302–317, Mar. 2021.

[23] H. Liang and W. Zhang, "Stochastic-Stackelberg-Game-Based Edge Service Selection for Massive IoT Networks," *IEEE Internet of Things J.*, vol. 10, no. 24, pp. 22 080–22 095, Dec. 2023.

[24] X. Chen, G. Zhu, H. Ding, L. Zhang, H. Zhang, and Y. Fang, "End-to-End Service Auction: A General Double Auction Mechanism for Edge Computing Services," *IEEE/ACM Trans. on Netw.*, vol. 30, no. 6, pp. 2616–2629, 2022.

[25] J. Yan, S. Bi, L. Duan, and Y.-J. A. Zhang, "Pricing-Driven Service Caching and Task Offloading in Mobile Edge Computing," *IEEE Trans. on Wireless Commun.*, vol. 20, no. 7, pp. 4495–4512, Feb. 2021.

[26] F. Tütüncüoğlu and G. Dán, "Optimal Service Caching and Pricing in Edge Computing: A Bayesian Gaussian Process Bandit Approach," *IEEE Trans. on Mobile Comput.*, vol. 23, no. 1, pp. 705–718, Jan. 2024.

[27] B. Simon, K. Keller, A. Sterz, B. Freisleben, O. Hinz, and A. Klein, "A Multi-Stakeholder Modeling Framework for the Techno-Economic Analysis of Telecommunication Networks," *IEEE Commun. Mag.*, vol. 61, no. 2, pp. 52–56, 2023.

[28] S. Pasteris, S. Wang, M. Herbster, and T. He, "Service Placement with Provable Guarantees in Heterogeneous Edge Computing Systems," in *Proc. of. IEEE Conf. on Computer Commun. (INFOCOM)*, 2019, pp. 514–522.

[29] S. Chen, L. Wang, and F. Liu, "Optimal Admission Control Mechanism Design for Time-Sensitive Services in Edge Computing," in *Proc. of IEEE Conf. on Comp. Commun. (INFOCOM)*, 2022, pp. 1169–1178.

[30] G. S. Dawson, R. T. Watson, and M.-C. Boudreau, "Information asymmetry in information systems consulting: Toward a theory of relationship constraints," *J. of Management Information Syst.*, vol. 27, no. 3, pp. 143–178, 2010.

[31] B. Jiang, L. Tian, Y. Xu, and F. Zhang, "To share or not to share: Demand forecast sharing in a distribution channel," *Marketing Science*, vol. 35, no. 5, pp. 800–809, 2016.

[32] M. Spence, "Job Market Signaling," *The Quarterly J. of Economics*, vol. 87, no. 3, pp. 355–374, Aug. 1973.

[33] K. Toczé and S. Nadjm-Tehrani, "A taxonomy for management and optimization of multiple resources in edge computing," *Wireless Commun. and Mobile Comput.*, vol. 2018, 2018.

[34] R. Chen, L. Li, R. Hou, T. Yang, L. Wang, and M. Pan, "Data-driven optimization for resource provision in non-cooperative edge computing market," in *Proc. of the IEEE Int. Conf. on Commun. (ICC)*, Jun. 2020, pp. 1–6.

[35] K. Chatterjee and W. Samuelson, "Bargaining under Incomplete Information," *Operations research*, vol. 31, no. 5, pp. 835–851, 1983.

[36] J. E. Parco, "Price-setting power and information asymmetry in sealed bidding," *Managerial and Decision Economics*, vol. 27, no. 6, pp. 413–434, 2006.

[37] J. Lähteenmäki, H. Hämmäinen, N. Zhang, and M. Swan, "Cost Modeling of a Network Service Provider Cloud Platform," in *Proc. of IEEE Int. Conf. on Cloud Engineering Workshop (IC2EW)*, 2016, pp. 148–153.

[38] A. AWS. (2023) Amazon ec2 – on-demand pricing. [Online]. Available: https://aws.amazon.com/ec2/pricing/on-demand/

[39] ——. (2023) Amazon ebs pricing. [Online]. Available: https://aws.amazon.com/ebs/pricing

[40] A. Mehta, W. Tärneberg, C. Klein, J. Tordsson, M. Kihl, and E. Elmroth, "How Beneficial Are Intermediate Layer Data Centers in Mobile Edge Networks?" in *Proc. of IEEE Int. Worksh. on Foundations and Applications of Self Systems (FASW)*, 2016, pp. 222–229.

[41] L. Gao, G. Iosifidis, J. Huang, L. Tassiulas, and D. Li, "Bargaining-Based Mobile Data Offloading," *IEEE J. on Selected Areas in Commun.*, vol. 32, no. 6, pp. 1114–1125, 2014.

[42] G. Premsankar and B. Ghaddar, "Energy-Efficient Service Placement for Latency-Sensitive Applications in Edge Computing," *IEEE Internet of Things J.*, vol. 9, no. 18, pp. 17 926–17 937, 2022.

[43] L. Wang, L. Jiao, T. He, J. Li, and H. Bal, "Service Placement for Collaborative Edge Applications," *IEEE/ACM Trans. on Netw.*, vol. 29, no. 1, pp. 34–47, 2021.

[44] J. F. Nash, "The Bargaining Problem," *Econometrica*, vol. 18, no. 2, pp. 155–162, 1950.

[45] H. Boche and M. Schubert, "Nash Bargaining and Proportional Fairness for Wireless Systems," *IEEE/ACM Trans. on Netw.*, vol. 17, no. 5, pp. 1453–1466, 2009.

[46] M. A. Satterthwaite and S. R. Williams, "Bilateral trade with the sealed bid k-double auction: Existence and efficiency," *J. of Economic Theory*, vol. 48, no. 1, pp. 107–133, 1989.

[47] W. Leininger, P. B. Linhart, and R. Radner, "Equilibria of the sealed-bid mechanism for bargaining with incomplete information," *J. of Economic Theory*, vol. 48, no. 1, pp. 63–106, 1989.

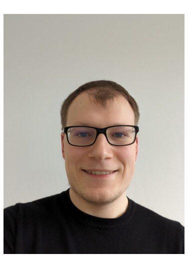
[48] T. Hou, G. Feng, S. Qin, and W. Jiang, "Proactive content caching by exploiting transfer learning for mobile edge computing," *Int. J. of Commun. Syst.*, vol. 31, no. 11, p. e3706, 2018.

[49] A. Abhari and D. Pudasaini, "Video analytic data reduction model for fog computing," in *Annual Modeling and Simulation Conf. (ANNSIM)*. IEEE, 2022, pp. 851–862.

[50] G. Carvalho, B. Cabral, V. Pereira, and J. Bernardino, "Edge computing: current trends, research challenges and future directions," *Computing*, vol. 103, pp. 993–1023, 2021.

[51] W. Gong, B. Zhang, and C. Li, "Task Assignment in Mobile Crowdsensing: Present and Future Directions," *IEEE Netw.*, vol. 32, no. 4, pp. 100–107, 2018.

[52] K. Velasquez, D. P. Abreu, M. Curado, and E. Monteiro, "Service Placement for Latency Reduction in the Fog Using Application Profiles," *IEEE Access*, vol. 9, pp. 80 821–80 834, 2021.



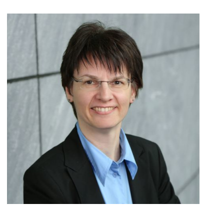
Bernd Simon (S'21) received the B.Sc. and M.Sc. degree in electrical engineering and information technology from Karlsruhe Institute of Technology, Germany, in 2015 and 2019, respectively. Currently, he is pursuing his Ph.D. in electrical engineering at Technische Universität Darmstadt, Germany, as a member of the Communications Engineering Laboratory. He is also part of the Collaborative Research Centre (CRC) 1053 "MAKI – Multi-Mechanisms Adaptation for the Future Internet" funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation). His research interests include wireless communications, game theory and machine learning with focus on resource management in multi-access edge computing (MEC) and mobile crowdsensing (MCS).

Paul Adrian received the B.Sc. and M.Sc. degree in Business Administration and Engineering with the specialization Electrical Engineering from Technische Universität Darmstadt, Germany, in 2020 and 2022, respectively. His research interests include the application of game theory for the resource management in multi-access edge computing (MEC).

Patrick Weber received the B.Sc. and M.Sc. degree in information systems from Technische Universität Darmstadt, Germany, in 2016 and 2019, respectively. Currently, he is pursuing his Ph.D. in Information Systems at the Goethe University Frankfurt, Germany, as a member of the Chair of Information Systems and Information Management. He is also part of the Collaborative Research Centre (CRC) 1053 "MAKI – Multi-Mechanisms Adaptation for the Future Internet" funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation). His research interests include information asymmetries and explainable artificial intelligence.

Patrick Felka is a Post-Doctoral Researcher at the Chair of Information Systems and Information Management, Goethe University Frankfurt, and as part of the Collaborative Research Centre (CRC) 1053 "MAKI—Multi-Mechanisms Adaptation for the Future Internet" funded by the Deutsche Forschungsgemeinschaft (DFG—German Research Foundation). He is conducting research on consumer behavior, data mining, and context-aware applications.

Oliver Hinz is currently a Professor of the Information Systems and Information Management, Goethe University Frankfurt. He is interested in research at the intersection of technology and markets. His research has been published in journals such as Information Systems Research, Management Information Systems Quarterly, Marketing Science, Journal of Marketing, Journal of Management Information Systems, Decision Support Systems, and Business & Information Systems Engineering, and in a number of proceedings such as ICIS, ECIS, and PACIS.



Anja Klein (M'96) received the Diploma and Dr.-Ing. (Ph.D.) degrees in electrical engineering from the University of Kaiserslautern, Germany, in 1991 and 1996, respectively. In 1996, she joined Siemens AG, Mobile Networks Division, Munich and Berlin. She was active in the standardization of third generation mobile radio in ETSI and in 3GPP, for instance leading the 3GPP RAN1 TDD group. She was director of a development department and a systems engineering department. In 2004, she joined the Technische Universität Darmstadt, Germany, as Full Professor, heading the Communications Engineering Laboratory. Her main research interests are in wireless networks, including resource management, interference management, UAV communications, edge computing, smart caching, energy harvesting, and cyber-physical systems, using methods from optimization, online learning, and game theory. Dr. Klein has authored over 400 peer-reviewed papers and has contributed to 12 books. She is inventor and co-inventor of more than 45 patents in the field of mobile communications. In 1999, she was named the Inventor of the Year by Siemens AG.