Multi-Stakeholder Service Placement via Iterative Bargaining With Incomplete Information

Artur Sterz, Patrick Felka, Bernd Simon, *Graduate Student Member, IEEE*, Sabrina Klos, *Member, IEEE*, Anja Klein, *Member, IEEE*, Oliver Hinz, and Bernd Freisleben, *Member, IEEE*

***Abstract*— Mobile edge computing based on cloudlets is an emerging paradigm to improve service quality by bringing computation and storage facilities closer to end users and reducing operating cost for infrastructure providers (IPs) and service providers (SPs). To maximize their individual benefits, IP and SP have to reach an agreement about placing and executing services on particular cloudlets. We show that a Nash Bargaining Solution (NBS) yields the optimal solution with respect to social cost and fairness if IP and SP have complete information about the parameters of their mutual cost functions. However, IP and SP might not be willing or able to share all information due to business secrets or technical limitations. Therefore, we present a novel iterative bargaining approach without complete mutual information to achieve substantial cost reductions for both IP and SP. Furthermore, we investigate how different degrees of information sharing impact social cost and fairness of the different approaches. Our evaluation based on the mobile augmented reality game Ingress shows that our approach achieves up to about 82% of the cost reduction that the NBS achieves and a cost reduction of up to 147% compared to traditional Take-it-or-Leave-it approaches, despite incomplete information.**

***Index Terms*—Edge computing, cloudlets, service placement, network economics.**

I. Introduction

MOBILE Edge Computing (MEC), i.e., providing computation and storage facilities at the edge of the Internet, has emerged as a new computing paradigm to improve latency-sensitive services like mobile augmented reality (AR) games, autonomous driving or high-quality live streaming, where centralized cloud computing approaches may not provide sufficient service quality due to high latency or low bandwidth [1], [2]. Apart from improving service quality, MEC can also reduce operational cost of networks [3], since the amount of communication from the edge through the network core can be reduced, e.g., in online gaming contexts up to 95% [3].

To achieve the goals of improved service quality and reduced cost, the use of *cloudlets* has been proposed [4], which are small-scale datacenters located at the edge of the Internet. Cloudlets provide computing resources accessible by users within a one-hop wireless connection. Services can be placed on cloudlets in addition to the cloud to reduce the distance between users and computing resources. By placing a service on a cloudlet, the service on the cloudlet will handle requests from connected users within a one-hop range. Otherwise, the cloud handles all other requests.

We consider the following stakeholders of the Internet: (i) infrastructure provider (IP), (ii) service provider (SP), (iii) cloud provider, and (iv) users. The IP (e.g., AT&T, Vodafone, etc.) owns and operates wireless access networks consisting of base stations (BSs), the network backend, as well as the MEC infrastructure such as cloudlets. The SP (e.g., Netflix, Niantic, etc.) offers a service, such as a mobile AR game or a high-quality video stream, to users. The SP relies on the availability of computational resources in the network, and the amount of required computational resources is proportional to the activity and the number of users. The cloud provider offers computational resources located at the core of the network. Users access the services provided by the SP that are either placed on the IP's cloudlets or the cloud provider's cloud.

Reducing network traffic is beneficial for both SP and IP, since this reduction results in lower cost for cloud services for the SP, and lower operational cost for the IP's network. Hence, for cost-optimal service placement, the SP and the IP are equally important. It is essential to involve both stakeholders to decide on which cloudlet a service should be deployed. While traditional cost models like Take-it-or-Leave-it may be suitable to optimize the cost of one stakeholder, cooperative decisions between SP and IP are promising to achieve better results in terms of social cost, i.e., the combined cost of both stakeholders compared to non-cooperative approaches, which in turn will improve the long-term cost reduction.

The question is how the IP should chose the cost for using cloudlets if the IP has no information about the SP's service usage. If chosen too high, the SP will not deploy any service

Manuscript received 19 July 2021; revised 7 January 2022; accepted 15 February 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor V. Aggarwal. Date of publication 15 March 2022; date of current version 18 August 2022. This work was supported in part by the German Research Foundation (DFG) through the Sonderforschungsbereich [Collaborative Research Center (SBF)] 1053 Multi-Mechanismen-Adaption für das künftige Internet [Multi-Mechanisms Adaption for the Future Internet (MAKI)] under Project 210487104; in part by the Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz (LOEWE) Program, Hessen, Germany (emergenCITY, Natur 4.0); and in part by the Bundesministerium für Bildung und Forschung [Federal Ministry of Education and Research Germany (BMBF)] Project Open6GHub under Grant 16KISK014. *(Corresponding author: Artur Sterz.)*

Artur Sterz and Bernd Freisleben are with the Department of Mathematics and Computer Science, Philipps University of Marburg, 35037 Marburg, Germany (e-mail: sterz@informatik.uni-marburg.de; freisleb@informatik.uni-marburg.de).

Patrick Felka and Oliver Hinz are with the Department of Information Systems and Information Management, Goethe University Frankfurt, 60323 Frankfurt, Germany (e-mail: felka@wiwi.uni-frankfurt.de; hinz@wiwi.uni-frankfurt.de).

Bernd Simon, Sabrina Klos, and Anja Klein are with the Communications Engineering Laboratory, Technical University of Darmstadt, 64283 Darmstadt, Germany (e-mail: b.simon@nt.tu-darmstadt.de; s.klos@nt.tu-darmstadt.de; a.klein@nt.tu-darmstadt.de).

Digital Object Identifier 10.1109/TNET.2022.3157040

TABLE I
OVERVIEW OF RELATED WORK

Contributions \ Related Work	[1]	[2]	[3], [4]	[5], [6]	[7]–[10]	[11]	[12], [13]	[14]	[15]	[16]	Our Approach
Cloudlet-based MEC	✓	✓	✓	✓	✓	✓	✓	-	-	-	✓
Considers IP and SP	✓	✓	-	-	-	-	-	✓	-	-	✓
Incomplete information	-	-	✓	✓	-	-	-	✓	✓	-	✓
Cost-awareness	✓	✓	✓	✓	✓	-	-	✓	✓	✓	✓
Evaluation using real data set	✓	-	-	✓	-	✓	-	✓	-	✓	✓

on the IP's cloudlets and the IP will miss out on possible cost reductions through reduced network traffic and a potentially better user experience for its customers. If chosen too low, the IP's deployment cost may exceed the potential cost reduction by the network traffic reduction. However, while IPs and SPs aim to reduce their individual cost, the parameters of their corresponding cost functions differ and thus the service placement on cloudlets depends on the individual stakeholder's perspective [21].

We propose a novel bargaining approach in which the IP and SP bargain for cost-effective service placements. Furthermore, both stakeholders have to incorporate incomplete information about the other stakeholder in the bargaining, since they are unwilling or not able to share all their information due to business secrets or technical limitations.

The main contributions of the paper are as follows:

- We propose a novel solution to the service placement problem in which SP and IP have to agree on the service placement and a cost for placing a service.
- We model the cost functions of SP and IP as a basis for finding a solution to the service placement problem.
- Using the proposed cost functions, we show for the first time that a Nash Bargaining Solution (NBS) exists, if both stakeholders have complete information about their mutual cost functions.
- Since in reality the stakeholders do not know each other's cost functions, we present a novel iterative bargaining approach to find a nearly optimal solution for both stakeholders.
- We investigate how different degrees of information sharing impact social cost reduction and fairness.
- We evaluate the proposed iterative bargaining approach using a real world data set of the popular mobile AR game Ingress. The results show that despite incomplete information, our approach achieves, on the average, up to 82% of the optimal solution's cost reduction, and about two times more game actions are processed on cloudlets compared to traditional Take-it-or-Leave-it models.

The remainder of the paper is organized as follows. Section II reviews related work. In Section III, we introduce our service placement scenario and system model. Section IV analyzes the bargaining process with complete information, and Section V presents our solution to the bargaining problem with incomplete information including two variants to discuss different scenarios of incomplete information. Experimental results based on our collected real-world data set are provided in Section VI. Finally, Section VII concludes the paper and outlines areas for future research.

II. RELATED WORK

Service placement is considered in several publications, where cloudlet-based MEC takes a large part in recent years, but also other areas like placing services on data caches or edge servers have been investigated.

Table I presents an overview of related work on service placement and considers aspects of existing approaches compared to our approach, such as whether the papers look at both IP and SP, whether they handle incomplete information, whether they present cost-aware solutions or try to optimize other aspects, and finally whether they use real-world data sets to evaluate their approaches.

Related work in the area of service placement often only focuses on one stakeholder (e.g., only IP or SP), or focuses only on the users. However, it is crucial to understand that stakeholders have different cost and different information about, e.g., the network, available resources or service usage. Therefore, it is important to consider both stakeholders in order to optimize social cost under realistic assumptions. The very few papers that focus on both stakeholders, however, do not handle incomplete information, but rely on complete information sharing between the two players. For example, the work of Gedeon *et al.* [5] requires all stakeholders not just to know where possible cloudlet locations are, but also what kind of hardware they use, how many resources are available, and how many users are likely to use a particular cloudlet. Similarly, in the work of Shih *et al.* [6], the SP shares its entire knowledge about its users (e.g., numbers and locations) with the IP, so that they can bargain for the price, even if stakeholders in these markets are usually not willing or not able to fully share information. The work of Cao *et al.* [18] seems to offer a solution that incorporates multiple stakeholders and incomplete information. However, the authors do not try to optimize the cost for both stakeholders, but only for the IP, while the SP has to either accept the IP's decision or decline it, but they do not interact with each other to find the best solution for both.

The literature on handling incomplete information usually does not take multiple stakeholders into account, but mainly tries to solve problems from one perspective only. For example, the work of Nguyen *et al.* [7] optimizes the IP's net profit and Zhan *et al.* [10] try to optimize the IP's operational cost in conjunction with a best achievable user experience. Chen *et al.* [8] only incorporate collaboration between different IPs. Finally, Ouyang *et al.* [19] base their placement strategy on information that is only available to the user.

Furthermore, both stakeholders, IP and SP, do not only try to improve their users' service experience, but also try to

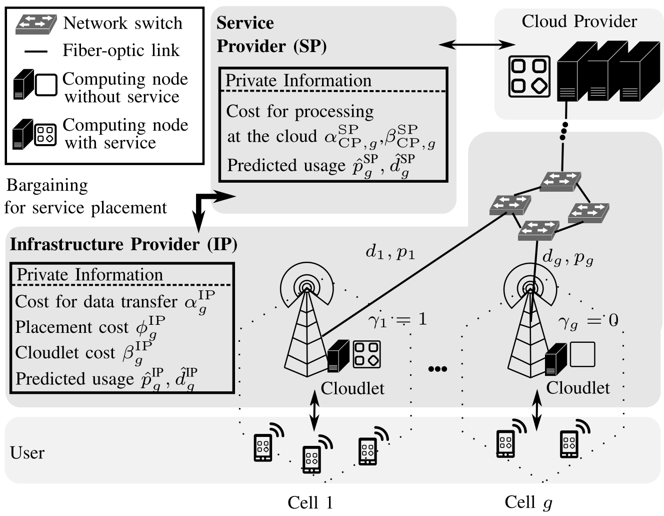


Fig. 1. Overview of the system model.

minimize their operational cost. Thus, it is essential to model the individual cost for both stakeholders and optimize the cost reductions for both stakeholders at the same time. However, most previous works mainly focus on optimizing the users' service experience, like Jia *et al.* [15], Mukherjee *et al.* [16] and Peng *et al.* [17], and do not include the stakeholders' cost at all. Jia *et al.* [15] offload the workload from the user's device to a cloudlet, Mukherjee *et al.* [16] reduce power consumption and latency, and Peng *et al.* [17] maximize the number of served users per cloudlet. Apart from that, the work of Liang *et al.* [9] tries to optimize the service placement problem by optimizing the placement locations so that the IP's budget is not exceeded.

Finally, large parts of related work evaluate their approaches based on synthetic data and assumptions that often are not based on real-world data sets. For example, Liu *et al.* [11], Ren *et al.* [13] and Yang *et al.* [14] use randomly generated data within ranges that are not motivated by literature or other sources. The work of Mondal *et al.* [12] models an average Australian city, but not based on a real-world data set, but on hypothetical assumptions. The same also applies to other papers cited above, as shown in Table I. While evaluations based on such data and assumptions are a good starting point to show the general feasibility, evaluations using real-world data sets help to underpin the practicality and applicability of theoretical approaches.

III. System Model

Fig. 1 provides an overview of our system model. In this scenario, four stakeholders interact with each other: service provider (SP), infrastructure provider (IP), cloud provider, and users. However, we only consider IP and SP as active parts, since only these two stakeholders have to jointly find suitable cloudlets for placing services on them. Users initiate service requests whenever they use a functionality of a service. Users want a high Quality-of-Experience (QoE) of a service, which is achieved by handling service requests on nearby cloudlets. A cloud serves as the default service placement location, where all service requests can be handled. Thus, users and the cloud provider are not active parts in our system model.

The IP operates base stations (BS) and a network backend, offers network access via its BSs, and deploys and operates permanently available cloudlets. The cloud is accessible via the IP's network backend. Furthermore, BSs provide ubiquitous radio coverage and network access for users, and multiple users can be connected to one BS. By default, the service is deployed in the cloud with sufficient capacity to handle the requests of all of its users.

The SP may rent resources on the IP's cloudlets and place the service on them by paying a monetary compensation. The service is placed on the cloudlet by offloading relevant parts of the service from the cloud to the cloudlet. If the service is not placed, e.g., because the SP and IP cannot agree on a cost, it remains in the cloud.

Furthermore, we assume that the network can be divided into a grid G of cells with one cloudlet per cell $g \in G$. For each cell $g \in G$, we introduce an indicator variable γ_g, which is 1 if the service is placed on the cloudlet in cell g and 0 otherwise. The goal is to place the service on particular cloudlets that are advantageous for both IP and SP in terms of cost reductions.

A. Service Provider

The SP provides a service to users which requires computation and data transfer between the users and the cloud. There are two different options how the service requests can be handled, one of which is fixed for each cell g. First, the service requests are handled by the cloud. After the service requests arrive at the BS, the data of size $\tilde{d}_g$ data units is transferred from the BS to the cloud. The computation is done in the cloud, which requires p_g processing units.

The second option is that the service requests are handled by a cloudlet in cell g. After the service requests arrive at the BS in cell g, they are handled by the cloudlet located in cell g, which requires p_g processing units on the cloudlet. In addition to the cloudlet, the SP needs to use cloud resources and the IP's backbone network, e.g., in mobile AR games for synchronizing local game states between the cloudlet and the global state in the cloud. The required data transfer between cloudlet and cloud is denoted as d_g. The amount of processing units required for this case in the cloud is reduced to $\tilde{p}_g < p_g$.

In the following, the cost models for the two options, using cloud resources or service placement on cloudlets, are introduced. The cost for providing the service on cloud resources ($\gamma_g = 0$) in cell g consists of two parts: the cost for data transfer and the cost for computing. First, the SP has to pay a cost factor $\alpha^{\mathrm{SP}}_{\mathrm{CP},g}$ for utilizing the IP's network for each of the $\tilde{d}_g$ data units that need to be transmitted from the BS to the cloud. Second, the SP has to pay a cost factor $\beta^{\mathrm{SP}}_{\mathrm{CP},g}$ to the cloud provider for the amount p_g of processing units required in the cloud, resulting in the following cost function:

$$\tilde{\kappa}^{\mathrm{SP}}_g = (1 - \gamma_g)\left[\alpha^{\mathrm{SP}}_{\mathrm{CP},g}\tilde{d}_g + \beta^{\mathrm{SP}}_{\mathrm{CP},g} p_g\right] \quad (1)$$

Note that the SP does not pay any cost for utilizing bandwidth from the users' devices to the BS, since these transmissions

TABLE II
MATHEMATICAL NOTATIONS

Notation	Description	Notation	Description
$g \in G$	Cell g in the grid G	α_g^{IP}	Cost factor for every data unit incurred for the IP in cell g
$\gamma_g \in \{0,1\}$	Indicator variable for placement in cell g. $\gamma_g = 1$ if a service is placed in g, zero otherwise	$\alpha_{\mathrm{CP},g}^{\mathrm{SP}}$	Cost factor for every data unit the SP has to pay to the cloud in cell g
$\kappa^{\{\mathrm{SP,IP}\}}$	Total cost for SP or IP	$\beta_{\mathrm{CP},g}^{\mathrm{SP}}$	Cost factor for every processing unit the SP has to pay to the cloud in cell g
$\kappa_g^{\{\mathrm{SP,IP}\}}$	Cost incurred for SP or IP if the cloudlet is used in cell g	β_g^{IP}	Cost factor for every processing unit incurred for the IP in cell g
$\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$	Cost that the SP has to pay to the IP for using the cloudlet in cell g	d_g	Data units from the cloudlet to the cloud (and vice versa) if service is placed in cell g
$\tilde{\kappa}_g^{\{\mathrm{SP,IP}\}}$	Cost incurred for SP or IP if no cloudlet is used in cell g	$\tilde{d}_g$	Data units from the BS to the cloud (or vice versa) if no service is placed in g
ϕ_g^{IP}	Fixed cost for placing a service on a cloudlet in cell g incurred for the IP	p_g	Processing units (i) in the cloudlet in cell g if service is placed in cell g or (ii) in the cloud in cell g if no service is placed in cell g
$\tilde{p}_g$	Processing units in the cloud in cell g even though service is placed in cell g		

need to be made regardless of whether cloudlets are used or not. Therefore, this factor is not considered in our cost model.

The cost for providing the service placed on a cloudlet ($\gamma_g = 1$) in cell g consists of three parts: the cost for data transfer, the cost for computing resources in the cloud, and the payment between SP and IP for the service placement. The cost for data transfer contains the cost factor $\alpha_{\mathrm{CP},g}^{\mathrm{SP}} \geq 0$ for every data unit d_g that is transferred between the cloudlet and cloud. The cost factor $\beta_{\mathrm{CP},g}^{\mathrm{SP}} \geq 0$ denotes the cost of each processing unit in the cloud. The payment between SP and IP for the service placement on a cloudlet is denoted by $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$. $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ consists of multiple parts, e.g., the cost incurred for placing a service initially on a cloudlet as well as the processing and data transfer to the cloudlet.

The bargaining between the IP and the SP to agree on the payment $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ is discussed in Sections IV and V. The resulting cost function κ_g^{SP} of the SP in cases where the service is placed on a cloudlet in cell g is

$$\kappa_g^{\mathrm{SP}} = \gamma_g \left[\kappa_{\mathrm{IP},g}^{\mathrm{SP}} + \alpha_{\mathrm{CP},g}^{\mathrm{SP}} d_g + \beta_{\mathrm{CP},g}^{\mathrm{SP}} \tilde{p}_g\right]. \tag{2}$$

The cost function of the SP for providing the service in the whole grid G consists of two parts: (i) the cost for cells where the service is placed on a cloudlet and (ii) the cost for cells running the service in the cloud. Overall, the cost function κ^{SP} of the SP for the whole grid G is as follows:

$$\kappa^{\mathrm{SP}} = \sum_{g \in G} \left[\kappa_g^{\mathrm{SP}} + \tilde{\kappa}_g^{\mathrm{SP}}\right]. \tag{3}$$

The SP aims to reduce its cost κ^{SP} by placing the service on suitable cloudlets [22]. However, only cells where the cost for running the service on a cloudlet is lower than the cost for using the cloud should be used [20]. The SP's cost reduction is a result of the fact that less data has to be sent through the IP's network backend from the BS to the cloud, since the service request will be handled directly at the cloudlet. By optimized selection of the cells where the services are placed, the cost given by Eq. (3) is lower than the cost that the SP has to pay to the cloud provider in the case without any cloudlet due to less network usage in the IP's backend.

B. Infrastructure Provider

Cloudlets contribute to the IP's goal of reducing its cost [22] by reducing the network traffic in its network backend, since large parts of the service usage are handled directly by the cloudlet. However, the parameters of the IP's cost function differ from the parameters of the SP's cost function. While the SP has to pay the cloud provider or the IP, depending on using the cloud or cloudlets, the IP has to pay for establishing and maintaining the network. Cloudlets reduce this cost by adding computing and storage resources at the edge network and reducing the traffic in the core network, which leads to a cost reduction for the IP. To reduce as much cost as possible, a cloudlet should serve many users, otherwise the deployment cost could be higher than the cost reduction from the decreased network usage.

We distinguish between three components in the cost of the IP: The fixed cost, which is required for operating the infrastructure for deployment and maintenance of the service, the cost for processing on the cloudlet, and the cost for transferring data over the IP's backend network.

The fixed cost ϕ_g^{IP} of the IP related to cell g contains all the infrastructure cost associated with deployment and maintenance of a service, which is assumed to be independent of the number of service requests in cell g. The cost for deploying the service, i.e., transferring the required data from the cloud to the cloudlet over the backend, only has to be paid if the service is not already deployed on the cloudlet in cell g. We do not consider service migration between different cloudlets, but we assume that a service is always offloaded from the cloud to the cloudlet. The maintenance of a service, e.g., regular updates and monitoring, incurs a cost for processing and data transfer for the IP (although less compared to cells where the service is not deployed), even if the service is already deployed on the cloudlet in cell g.

The second component of the cost, the cost for processing on the cloudlet, is modeled in each cell g by the cost factor β_g^{IP} per processing unit. The total cost for processing is given by the cost factor β_g^{IP} multiplied by the number p_g of processing units required on the cloudlet in the given cell g.

The third component of the cost is the cost for transferring data over the IP's backend network. For each data unit transmitted over the backend network, a cost factor for α_g^{IP} is incurred. The total cost for data transfer is given by the number of data units d_g multiplied with the cost factor α_g^{IP}.

To make service placement on cloudlets profitable for the IP, the IP has to receive a payment by the SP, which is larger than the aforementioned cost. The SP pays $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ to the IP for placing the service on the cloudlet in cell g. How the payment $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ is determined such that is acceptable for both the IP and the SP, is the central problem discussed in the remainder of this work.

Combining all the components of the cost and the payment of the SP, results in the following cost for the IP if a cloudlet is used in $g \in G$:

$$\kappa_g^{\mathrm{IP}} = \gamma_g \left[\phi_g^{\mathrm{IP}} + \beta_g^{\mathrm{IP}} p_g + \alpha_g^{\mathrm{IP}} d_g - \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right] \tag{4}$$

If no cloudlet is used in cell g (i.e., $\gamma_g = 0$), the IP only has the cost factor α_g^{IP} for utilizing the backend network to transfer each of the $\tilde{d}_g$ data units from the cloudlet to the cloud, leading to the following cost function:

$$\tilde{\kappa}_g^{\mathrm{IP}} = (1 - \gamma_g) \left[\alpha_g^{\mathrm{IP}} \tilde{d}_g\right] \tag{5}$$

This results in the following cost for the IP for the grid G:

$$\kappa^{\mathrm{IP}} = \sum_{g \in G} \left[\kappa_g^{\mathrm{IP}} + \tilde{\kappa}_g^{\mathrm{IP}}\right] \tag{6}$$

Note that the IP does not have to pay the cost given in Eq. (6) directly to any stakeholder, but indirectly, e.g., through higher investments into the network to handle the amount of data from all services, which can be summarized as operating expenses. The cost for deploying a cloudlet and letting SPs place services on it will be paid by the SP to the IP to some degree, which results in a cost reduction.

Table II summarizes the mathematical notation and gives an overview of the used symbols.

C. Resource Limitations and Competition

We assume that sufficient resources are available at each cloudlet to serve the SP's demand for processing units p_g in cell g. In cases where the computational demand of the service would exceed the resources of the cloudlet, we assume that ϕ_g^{IP} (i.e., the cost for placing a service on a cloudlet incurred to the IP due to the initial data transfer) contains additional cost to scale the provided hardware in cell g accordingly.

This implies that although either a single SP may offer multiple services or multiple SPs may exist, our approach presented in Section V can be applied to each service separately, since the IP can handle unlimited services without causing conflict situations between SPs. Thus, we do not consider multiple SPs explicitly in the remainder of this paper.

Finally, each cell g may have cloudlets from different IPs. However, we consider only a single IP in the remainder of the paper without loss of generality, since our approach is applicable to each IP in a cell separately to find the IP with the highest cost reduction per cell. If multiple cloudlets from different IPs are available in one cell, the cost function applies to each IP. Furthermore, the SP may also opt to place the service on cloudlets of different IPs in one cell if the calculated cost reduction exceeds the cost in the cloud for that cell.

D. Problem Formulation

For each cell $g \in G$, the SP and IP need to come to an agreement whether the service should be placed on the cloudlet in cell g (i.e., $\gamma_g = 1$) or not (i.e., $\gamma_g = 0$). In the first case, they also need to agree on a cost $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ to be paid by the SP to the IP. If they disagree, no service is placed in cell g (i.e., $\gamma_g = 0$, $\kappa_{\mathrm{IP},g}^{\mathrm{SP}} = 0$). To reach an agreement, a bargaining solution is needed that satisfies the following properties: (i) no participant of the bargaining should have disadvantages by participating in the bargaining (individual rationality) and (ii) the cost reduction caused by a service placement on cloudlets should be shared equally between the IP and SP (fairness). This gives a strong incentive for both the SP and the IP to participate in a bargaining for service placement. In the following, we derive a solution for service placement and the associated cost.

IV. Bargaining With Complete Information

In this section, we present our solution to the problem stated in Section III-D under the assumption that both the IP and SP know all parameters of their own cost function and the cost function of the other stakeholder. In this case, a Nash Bargaining Solution (NBS) can be used to find an optimal agreement between IP and SP.

Nash bargaining is a two-person bargaining framework [23], where two stakeholders either reach an agreement a from a set A of alternatives or fail to reach an agreement, in which case the bargaining ends at a disagreement point d. Each stakeholder $i \in \{1, 2\}$ has a utility function U_i over the set of agreements and the disagreement point. Nash showed that under mild technical conditions, there exists a unique bargaining solution, called Nash Bargaining Solution (NBS), which satisfies a set of four axioms that any plausible bargaining solution should satisfy [23]. It can be shown [6], [23], [24] that an agreement $a \in A$ is a NBS if it solves the following optimization problem:

$$\begin{aligned} &\max_{a \in A} \ (U_1(a) - U_1(d)) \cdot (U_2(a) - U_2(d)) \\ &\text{s.t. } U_1(a) \geq U_1(d), U_2(a) \geq U_2(d). \end{aligned} \tag{7}$$

Hence, the NBS maximizes the product of both stakeholders' utility gains compared to the disagreement outcome. In the following, for each fixed cell g, we compute the NBS for the problem stated in Section III-D.

A. Agreement Set and Disagreement Point

The set A_g of possible agreements between the IP and the SP in cell g is defined by

$$A_g := \left\{ \left(\gamma_g, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) | \gamma_g = 1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}} \in (-\infty, \infty) \right\} \tag{8}$$

Note that the price $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ may also be negative, since if placing the service in the cell is more profitable for the IP than the SP, the IP may have to pay the SP to reach an agreement.

If IP and SP do not come to an agreement for cell g, no service is placed in cell g. A disagreement in cell g is described by the following disagreement point:

$$\left(\gamma_g, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) = (0, 0) \tag{9}$$

B. Utilities of SP and IP

In Nash bargaining, each stakeholder has a utility function over the agreement set and the disagreement point. In our case, the utility functions per cell correspond to the negative cost functions per cell. Hence, according to Eq. (1) and Eq. (2), the utility of the SP for cell g is given by

$$\begin{aligned} U_g^{\mathrm{SP}}\left(\gamma_g, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) := \gamma_g \Big[&- \kappa_{\mathrm{IP},g}^{\mathrm{SP}} - \alpha_{\mathrm{CP},g}^{\mathrm{SP}} d_g - \beta_{\mathrm{CP},g}^{\mathrm{SP}} \tilde{p}_g \Big] \\ &+(1-\gamma_g)\Big[- \alpha_{\mathrm{CP},g}^{\mathrm{SP}} \tilde{d}_g - \beta_{\mathrm{CP},g}^{\mathrm{SP}} p_g \Big] \end{aligned} \tag{10}$$

and, according to Eq. (4) and Eq. (5), the utility of the IP for cell g is given by

$$\begin{aligned} U_g^{\mathrm{IP}}\left(\gamma_g, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) := \gamma_g \Big[&- \phi_g^{\mathrm{IP}} - \alpha_g^{\mathrm{IP}} d_g - \beta_g^{\mathrm{IP}} p_g + \kappa_{\mathrm{IP},g}^{\mathrm{SP}} \Big] \\ &+(1-\gamma_g)\Big[- \alpha_g^{\mathrm{IP}} \tilde{d}_g \Big]. \end{aligned} \tag{11}$$

C. Disagreement Outcome

The disagreement outcome is given by the utilities of the SP and IP, if they choose the disagreement point:

$$\left(U_g^{\mathrm{SP}}(0,0), U_g^{\mathrm{IP}}(0,0)\right) = \left(-\alpha_{\mathrm{CP},g}^{\mathrm{SP}} \tilde{d}_g - \beta_{\mathrm{CP},g}^{\mathrm{SP}} p_g, -\alpha_g^{\mathrm{IP}} \tilde{d}_g\right) \tag{12}$$

D. Feasible Agreement Points

The IP and SP will only accept an agreement $\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \in A_g$ if, for both of them, the agreement is better than the disagreement outcome.

An agreement $\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \in A_g$ is better for the SP than the disagreement outcome, if $U_g^{\mathrm{SP}}\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \geq U_g^{\mathrm{SP}}(0,0)$ holds, which by Eq. (10) and Eq. (12) is equivalent to

$$\kappa_{\mathrm{IP},g}^{\mathrm{SP}} \leq \alpha_{\mathrm{CP},g}^{\mathrm{SP}}(\tilde{d}_g - d_g) + \beta_{\mathrm{CP},g}^{\mathrm{SP}}(p_g - \tilde{p}_g) =: L_g^{\mathrm{SP}} \tag{13}$$

Hence, $L_g^{\mathrm{SP}} \geq 0$ gives an upper limit on the price that the SP would accept. The value L_g^{SP} corresponds to the cost reduction for the SP when a service is placed in cell g.

An agreement $\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \in A_g$ is better for the IP than the disagreement outcome, if $U_g^{\mathrm{IP}}\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \geq U_g^{\mathrm{IP}}(0,0)$ holds, which by Eq. (11) and Eq. (12) is equivalent to

$$\kappa_{\mathrm{IP},g}^{\mathrm{SP}} \geq \phi_g^{\mathrm{IP}} + \alpha_g^{\mathrm{IP}}(d_g - \tilde{d}_g) + \beta_g^{\mathrm{IP}} p_g =: l_g^{\mathrm{IP}} \tag{14}$$

Hence, l_g^{IP} gives a lower limit on the cost to be paid by the SP such that the IP would accept the agreement. The value l_g^{IP} corresponds to the cost increase that the IP experiences when the service is placed in cell g. Note that l_g^{IP} can also be negative ($l_g^{\mathrm{IP}} < 0$) if the IP has a cost reduction by placing the service in cell g even without a payment by the SP.

A feasible agreement point is an agreement $\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) \in A_g$ that satisfies Eq. (13) and Eq. (14). Such feasible agreement points do not necessarily exist. In detail, if $L_g^{\mathrm{SP}} < l_g^{\mathrm{IP}}$ holds, i.e., the SP's upper cost limit L_g^{SP} is lower than the IP's lower cost limit l_g^{IP}, by Eq. (13) and Eq. (14), no feasible agreement point exists. In this case, the IP and SP will choose the disagreement option $\gamma_g = 0$.

However, if $L_g^{\mathrm{SP}} \geq l_g^{\mathrm{IP}}$, each cost $\kappa_{\mathrm{IP},g}^{\mathrm{SP}}$ in the interval $\left[l_g^{\mathrm{IP}}, L_g^{\mathrm{SP}}\right]$ leads to a feasible agreement.

E. Nash Bargaining Solution

We now formulate the optimization problem according to Eq. (7) to compute the NBS. If feasible agreement points exist, i.e., if $L_g^{\mathrm{SP}} \geq l_g^{\mathrm{IP}}$ holds, the NBS is the optimal solution of the following problem:

$$\begin{aligned} &\max_{\kappa_{\mathrm{IP},g}^{\mathrm{SP}}} \ f(\kappa_{\mathrm{IP},g}^{\mathrm{SP}}) \\ &\text{s.t. } \kappa_{\mathrm{IP},g}^{\mathrm{SP}} \in [l_g^{\mathrm{IP}}, L_g^{\mathrm{SP}}] \end{aligned} \tag{15}$$

where

$$\begin{aligned} f(\kappa_{\mathrm{IP},g}^{\mathrm{SP}}) &:= \left(U_g^{\mathrm{SP}}\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) - U_g^{\mathrm{SP}}(0,0)\right) \\ &\quad \cdot \left(U_g^{\mathrm{IP}}\left(1, \kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right) - U_g^{\mathrm{IP}}(0,0)\right) \\ &= -\left(\kappa_{\mathrm{IP},g}^{\mathrm{SP}}\right)^2 + \kappa_{\mathrm{IP},g}^{\mathrm{SP}}(l_g^{\mathrm{IP}} + L_g^{\mathrm{SP}}) - l_g^{\mathrm{IP}} L_g^{\mathrm{SP}} \end{aligned} \tag{16}$$

To compute the NBS, we set the derivative to zero, i.e.,

$$0 = f'(\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}}) = -2\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}} + l_g^{\mathrm{IP}} + L_g^{\mathrm{SP}}, \tag{17}$$

we get

$$\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}} = \frac{1}{2}\left(l_g^{\mathrm{IP}} + L_g^{\mathrm{SP}}\right) \tag{18}$$

and since $f''(\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}}) = -2 < 0$, and $\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}} \in [l_g^{\mathrm{IP}}, L_g^{\mathrm{SP}}]$, the result in Eq. (18) gives the optimal solution of Eq. (15). Hence, if $L_g^{\mathrm{SP}} \geq l_g^{\mathrm{IP}}$ holds, according to the NBS, SP and IP will agree on the price $\kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}}$ in Eq. (18), which is the average of L_g^{SP} and l_g^{IP}. Intuitively, SP and IP equal out their cost, so that after the payment, both benefit from the same cost reduction.

V. Iterative Bargaining With Incomplete Information

In Section IV, we have presented the NBS solution for the case that the SP and the IP have complete information about their own cost function and the cost function of the bargaining partner. In this section, we consider two types of incomplete information in the system model: incomplete information about the cost factors of the bargaining partner and incomplete information about the service usage. Furthermore, we propose a novel iterative bargaining approach under incomplete information to overcome the challenges imposed by the incomplete information.

A. Nash Bargaining With Incomplete Information

We extend the Nash bargaining problem from Section IV to incorporate incomplete information for each stakeholder. The first type of incomplete information is concerning the cost factors of the bargaining partner. The IP does not know the cost factor $\alpha_{\mathrm{CP},g}^{\mathrm{SP}}$ for transferring data in the cloud and the cost factor $\beta^{\mathrm{SP}}_{\mathrm{CP},g}$ for processing in the cloud. The SP does not know the cost ϕ^{IP}_g for placing a service, the cost factor α^{IP}_g for transferring data over the network and the cost factor β^{IP}_g for processing on the cloudlet. This type of information is relevant to predict the potential cost savings of the bargaining partner in case of an agreement. The SP and IP typically do not want to share this private information with each other, therefore we need to consider this information as incomplete in the bargaining procedure.

The second type of incomplete information is concerning the service usage in the cell g. The IP has no information about the SP's users activity and context. Therefore, the prediction of service usage from the IP's perspective can only be based on other information. The service usage directly affects the cost function κ^{IP} of the IP. In particular, the change l^{IP}_g in the cost function by a service placement is not known by the IP.

To incorporate incomplete information about the service usage we present two different approaches: No Information Sharing (NIS) uses individual service usage predictions by the IP and the SP, and Partial Information Sharing (PIS), where the SP shares its service usage predictions with the IP.

B. Pre-Bargaining Information Acquisition and Sharing

1) Cost Factor Predictions: Before the bargaining procedure starts, the IP and the SP acquire information about the bargaining partner. It is possible to obtain predictions of the bargaining partner's cost factors within a range of possible values from publicly available sources, e.g., from price lists of competitors. Although some information about the bargaining partner can be obtained using publicly available sources, the available information on both sides is still incomplete, hence IP and SP cannot simply apply the NBS from Section IV.

The SP has to predict l^{IP}_g (Eq. (14)), which is the minimum price the IP would accept for placing the service on the cloudlet in cell g. We model the incomplete information as the uncertainty of the SP in l^{IP}_g. For this uncertainty from the perspective of the SP, we introduce a probability density function (PDF) of l^{IP}_g, which will be denoted as $p_{l^{\mathrm{IP}}_g}(l^{\mathrm{IP}}_g)$. We assume no further knowledge about the IP exists, leading to the assumption of a uniform distribution between a lower bound $l^{\mathrm{IP}}_{g,n,\min}$ and an upper bound $l^{\mathrm{IP}}_{g,n,\max}$, resulting in the PDF

$$p_{l^{\mathrm{IP}}_g}(l^{\mathrm{IP}}_g)=\begin{cases}\dfrac{1}{l^{\mathrm{IP}}_{g,n,\max}-l^{\mathrm{IP}}_{g,n,\min}}, & l^{\mathrm{IP}}_g\in[l^{\mathrm{IP}}_{g,n,\min},l^{\mathrm{IP}}_{g,n,\max}]\\ 0, & \text{else.}\end{cases} \tag{19}$$

The SP can obtain a lower bound $l^{\mathrm{IP}}_{g,n,\min}$, e.g., by predicting the cost for the additional hardware and energy of the IP for service placement [25], and an upper bound $l^{\mathrm{IP}}_{g,n,\max}$, e.g., by using public price lists for service placement [26].

The IP has to predict L^{SP}_g (Eq. (13)), which is the upper limit on the price that the SP would accept. Analogously to the cost factor prediction of the SP, we introduce for this uncertainty of the IP a PDF of L^{SP}_g, which will be denoted as $p_{L^{\mathrm{SP}}_g}(L^{\mathrm{SP}}_g)$. Furthermore, we assume that the IP can determine an interval for L^{SP}_g described by a lower bound $L^{\mathrm{SP}}_{g,n,\min}$ and an upper bound $L^{\mathrm{SP}}_{g,n,\max}$, resulting in the PDF

$$p_{L^{\mathrm{SP}}_g}(L^{\mathrm{SP}}_g)=\begin{cases}\dfrac{1}{L^{\mathrm{SP}}_{g,n,\max}-L^{\mathrm{SP}}_{g,n,\min}}, & L^{\mathrm{SP}}_g\in[L^{\mathrm{SP}}_{g,n,\min},L^{\mathrm{SP}}_{g,n,\max}]\\ 0, & \text{else.}\end{cases} \tag{20}$$

The IP can obtain the lower bound $L^{\mathrm{SP}}_{g,n,\min}$ by predicting the reduction in cloud and backhaul cost of the SP (e.g., [27]), and the upper bound $L^{\mathrm{SP}}_{g,n,\max}$ by using public price lists of cloud providers (e.g., [28], [29]). In the next step, the IP and the SP predict the service usage in each cell g of the grid G.

2) Service Usage Prediction: The SP can predict the service usage based on measurements of the users' activity, users' context and general information like weather conditions and population densities. The exact service prediction procedure of the SP will not be further discussed in this paper, since there are several proposals in the literature (e.g., [30], [31]). We model the result of the SP's prediction of the service processing requirements p_g and data d_g as

$$\hat{p}^{\mathrm{SP}}_g=p_g+n^{\mathrm{p,SP}}_g,\quad \hat{d}^{\mathrm{SP}}_g=d_g+n^{\mathrm{d,SP}}_g \tag{21}$$

where $n^{\mathrm{p,SP}}_g$ ($n^{\mathrm{d,SP}}_g$) is a Gaussian distributed random variable with a standard deviation of $\sigma^{\mathrm{p,SP}}_g$ ($\sigma^{\mathrm{d,SP}}_g$). The accuracy of the prediction, which is given by $\sigma^{\mathrm{p,SP}}_g$ and $\sigma^{\mathrm{d,SP}}_g$, depends on the cell g. The remaining data $\tilde{d}_g$ and service processing requests $\tilde{p}_g$ to the cloud are predicted analogously.

The IP has no information about the SP's users and therefore relies on contextual information in each cell g, such as weather conditions and population densities. We model the IP's prediction of the service processing requirements p_g and data d_g as

$$\hat{p}^{\mathrm{IP}}_g=p_g+n^{\mathrm{p,IP}}_g,\quad \hat{d}^{\mathrm{IP}}_g=d_g+n^{\mathrm{d,IP}}_g \tag{22}$$

where $n^{\mathrm{p,IP}}_g$ ($n^{\mathrm{d,IP}}_g$) is a Gaussian distributed random variable with a standard deviation of $\sigma^{\mathrm{p,IP}}_g$ ($\sigma^{\mathrm{d,IP}}_g$). Note that $\sigma^{\mathrm{p,IP}}_g$ and $\sigma^{\mathrm{d,IP}}_g$ may be substantially larger than $\sigma^{\mathrm{p,SP}}_g$ and $\sigma^{\mathrm{d,SP}}_g$ respectively, since the IP has no information about the SP's users. The prediction of the remaining data $\tilde{d}_g$ and service processing requests $\tilde{p}_g$ to the cloud are modelled analogously to Eq. (22).

In the next section, the sharing of the service usage predictions is discussed. For this procedure, two alternatives will be presented. The first one is without any information sharing between the stakeholders, whereas in the second one both stakeholders share their service usage prediction with each other.

3) No Information Sharing (NIS): This approach does not involve any information sharing between IP and SP before the bargaining procedure. Therefore, both stakeholders use a different service usage prediction to forecast their resulting cost of a service placement. To predict the IP's cost l^{IP}_g in case of a service placement in cell g, the IP considers the service usage prediction using Eq. (22). Since the IP knows its own cost factors, the IP can calculate l^{IP}_g by Eq. (14). The

SP can predict its potential cost savings L_g^{SP} (Eq. (13)) in case of a service placement with its service usage prediction according to Eq. (21).

4) Partial Information Sharing (PIS): In this approach, both stakeholders report their service usage predictions for cell g to the bargaining partner. In cases where the SP has no experience with deploying the service in cell g, the prediction of the IP could be more accurate. In cells with high service usage in the past, the prediction of the SP could be more accurate. Sharing the service usage predictions does not introduce negative consequences for both, since both can measure the service usage anyway. Both the IP and SP will use the more accurate prediction to calculate their own cost functions $l_g^{\mathrm{IP}}, L_g^{\mathrm{SP}}$.

Note that using the PIS variant for the upcoming iterative bargaining approach, one of the stakeholders, or both, could lie about the predicted service usage, whereas using the NIS variant, this is not possible. To prevent stakeholders from abusing wrong predictions of service usages, there are several mechanisms to enforce truthfulness in repeated games, e.g., reputation-based methods [32], so that we will not further investigate it in this paper. The remaining structure of our iterative bargaining approach is the same, regardless of whether NIS or PIS is used to acquire all required information and compute the IP's and SP's cost functions.

Using either its own prediction of d_g, $\tilde{d}_g$ and p_g or the values it got from the SP, the IP can predict the minimum price l_g^{IP} that should be paid by the SP. However, the IP has limited knowledge about the valuation L_g^{SP}, i.e., how profitable a service placement is for the SP. The SP, on the other hand, computes the maximum cost L_g^{SP} that it would accept for using the cloudlet. However, the SP has limited knowledge about the valuation l_g^{IP}, i.e., how profitable or expensive a service placement on a cloudlet in cell g is for the IP.

C. Iterative Bargaining

Our solution is an iterative bargaining approach with sealed offers, as shown in Fig. 2. The value n indicates the index of the current iteration.

1) Offering Phase: The SP offers a maximum acceptable cost $o_{g,n}^{\mathrm{SP}}$ to pay for using the cloudlet, whereas the IP offers a minimum acceptable cost $o_{g,n}^{\mathrm{IP}}$ for using the cloudlet. The SP and the IP update their offers according to their corresponding information in round n, i.e., the cost $o_{g,n}^{\mathrm{SP}}$ and $o_{g,n}^{\mathrm{IP}}$ may not correspond to their actual true valuations L_g^{SP} and l_g^{IP}.

2) Revealing Phase: Both offers are revealed simultaneously. If the SP's offer is higher than the offer of the IP $o_{g,n}^{\mathrm{SP}} > o_{g,n}^{\mathrm{IP}}$ the bargaining is finished. The final cost is calculated as

$$\kappa_{\mathrm{IP},g}^{\mathrm{SP}} = \frac{1}{2} \cdot (o_{g,n}^{\mathrm{SP}} + o_{g,n}^{\mathrm{IP}}). \tag{23}$$

Choosing the final cost as the average of the two offers has been shown to maximize the social cost [33]. If $o_{g,n}^{\mathrm{SP}} < o_{g,n}^{\mathrm{IP}}$ the bargaining will continue, since the SP's offer is lower than the minimum acceptable cost of the IP. If both repeat a previous offer, i.e. $o_{g,n}^{\mathrm{SP}} = o_{g,n-1}^{\mathrm{SP}}$, and $o_{g,n}^{\mathrm{IP}} = o_{g,n-1}^{\mathrm{IP}}$, neither SP nor IP are willing to give an offer closer to the acceptable region. They will disagree, the bargaining procedure will stop,

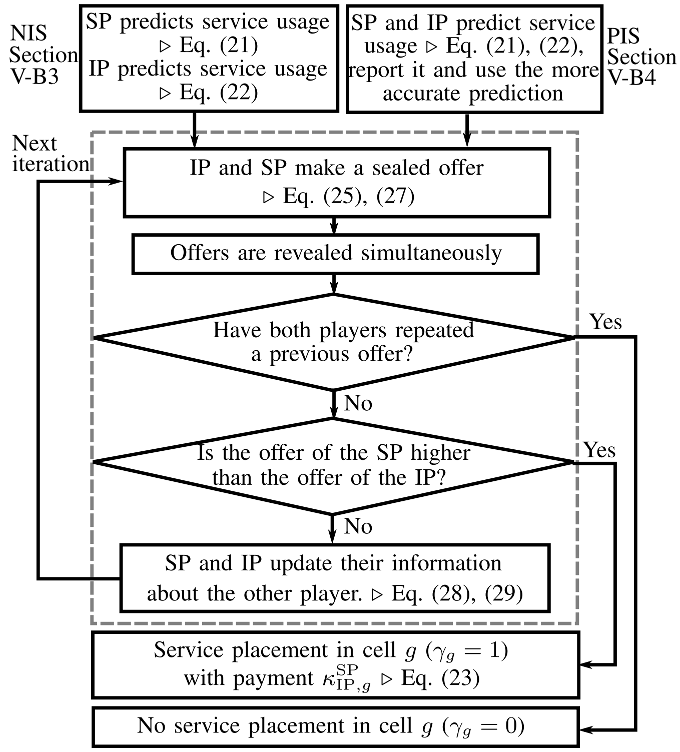


Fig. 2. The iterative bargaining procedure.

and the service will not be placed in this cell. Furthermore, we set a limit N for the number of iterations, such that the disagreement outcome is chosen if $n > N$.

3) Update Phase: After seeing the offer of the other stakeholder, both stakeholders update their knowledge about the valuation of the other stakeholder. Then, the next round $n+1$ of the bargaining process starts.

SP and IP only have access to incomplete information regarding the valuation of the other stakeholder and consequently do not know the worst case of the other stakeholder. Therefore, we derive an optimal bidding strategy for SP and IP under incomplete information. The expected profit of the SP considering the incomplete information about l_g^{IP} depending on its offer $o_{g,n}^{\mathrm{SP}}$ in cell g is

$$\begin{aligned}\pi_{g,\mathrm{SP}}(o_{g,n}^{\mathrm{SP}}) &= \mathbb{E}_{l_g^{\mathrm{IP}}}(L_g^{\mathrm{SP}} - \kappa_{\mathrm{IP},g}^{\mathrm{SP}}) \\ &= \int_{-\infty}^{o_{g,n}^{\mathrm{SP}}} \left(L_g^{\mathrm{SP}} - \frac{1}{2}\left(o_{g,n}^{\mathrm{SP}} + s \right)\right) \cdot p_{l_g^{\mathrm{IP}}}(s)\,\mathrm{d}s. \end{aligned} \tag{24}$$

Three edge cases for $o_{g,n}^{\mathrm{SP}}$ can be distinguished: (i) $o_{g,n}^{\mathrm{SP}} < l_{g,n,\min}^{\mathrm{IP}}$: the offer of the SP is smaller than the predicted lower bound for the cost of the IP, i.e., the expected profit (Eq. (24)) is zero, since the probability of an agreement is zero; (ii) $o_{g,n}^{\mathrm{SP}} > L_g^{\mathrm{SP}}$: the offer of the SP is higher than its own benefit from an agreement, i.e., the expected profit (Eq. (24)) may be negative; (iii) $o_{g,n}^{\mathrm{SP}} > l_{g,n,\max}^{\mathrm{IP}}$: the SP's offer is higher than the maximum predicted cost $l_{g,n,\max}^{\mathrm{IP}}$ of the IP, i.e., this offer leads to the probability of an agreement of 1.0, but is clearly suboptimal for the expected profit of the SP. The SP could decrease its offer and increase its expected profit.

We assume that the SP and the IP are risk-neutral, meaning that they maximize their individual expected utility. Furthermore, we assume individual rationality, i.e., a stakeholder only gives an offer if the expected utility is positive. The optimal offer $o_{g,n}^{\mathrm{SP}}$ for the SP in cell g is

$$\begin{aligned} o_{g,\mathrm{SP}}^{*,(t)} &= \max_{o_{g,n}^{\mathrm{SP}}} \quad \pi_{g,\mathrm{SP}}(o_{g,n}^{\mathrm{SP}}) \\ &= \begin{cases} \frac{2}{3}\cdot(L_g^{\mathrm{SP}}+\frac{1}{2}l_{g,n,\min}^{\mathrm{IP}}) & L_g^{\mathrm{SP}} < \frac{3}{2}l_{g,n,\min}^{\mathrm{IP}} \\ L_g^{\mathrm{SP}}, & L_g^{\mathrm{SP}} > \frac{3}{2}l_{g,n,\min}^{\mathrm{IP}} \end{cases} \end{aligned} \tag{25}$$

The expected profit of the IP considering the incomplete information about L_g^{SP} depending on its offer $o_{g,n}^{\mathrm{IP}}$ in cell g is

$$\begin{aligned} \pi_{g,\mathrm{IP}}(o_{g,n}^{\mathrm{IP}}) &= \mathbb{E}_{L_g^{\mathrm{SP}}}(\kappa_{\mathrm{IP},g}^{\mathrm{SP}} - l_g^{\mathrm{IP}}) \\ &= \int_{o_{g,n}^{\mathrm{IP}}}^{\infty} \left(\frac{1}{2}\left(o_{g,n}^{\mathrm{SP}}+s\right) - l_g^{\mathrm{IP}}\right)\cdot p_{L_g^{\mathrm{SP}}}(s)\,\mathrm{d}s. \end{aligned} \tag{26}$$

Three edge cases for $o_{g,n}^{\mathrm{IP}}$ can be distinguished: (i) $o_{g,n}^{\mathrm{IP}} < l_g^{\mathrm{IP}}$: the offer of the IP is smaller than its own cost induced by the service placement, i.e., the expected profit (Eq. (26)) may be negative; (ii) $o_{g,n}^{\mathrm{IP}} < L_{g,n,\min}^{\mathrm{SP}}$: in this case, the probability of an agreement is 1.0, since the IP offers less than the minimum predicted cost reduction of the SP, i.e., this type of offer is suboptimal, since the IP could increase $o_{g,n}^{\mathrm{IP}}$ and increase its expected profit; (iii) $o_{g,n}^{\mathrm{IP}} > L_{g,n,\max}^{\mathrm{SP}}$: the offer of the IP is higher than the expected cost reduction at the SP, i.e., the expected profit (Eq. (26)) is zero, since the probability of an agreement is zero. The optimal offer $o_{g,n}^{\mathrm{SP}}$ for the IP in cell g is

$$\begin{aligned} o_{g,\mathrm{IP}}^{*,(t)} &= \max_{o_{g,n}^{\mathrm{SP}}} \quad \pi_{g,\mathrm{SP}}(o_{g,n}^{\mathrm{SP}}) \\ &= \begin{cases} l_g^{\mathrm{IP}}, & l_g^{\mathrm{IP}} > \frac{3}{2}L_{g,n,\max}^{\mathrm{SP}} \\ \frac{2}{3}\cdot(l_g^{\mathrm{IP}}+\frac{1}{2}L_{g,n,\max}^{\mathrm{SP}}) & l_g^{\mathrm{IP}} < \frac{3}{2}L_{g,n,\max}^{\mathrm{SP}}. \end{cases} \end{aligned} \tag{27}$$

After seeing the offers of the other stakeholder, both decide to update their information about the valuation of the other stakeholder. For this update, we choose a sequential linear estimator with the adjustment rates λ^{SP}, λ^{IP} based on the observation of the offer of the other stakeholder:

$$l_{g,n+1,\min}^{\mathrm{IP}} = l_{g,n,\min}^{\mathrm{IP}} + \lambda^{\mathrm{SP}}\cdot(o_{g,n}^{\mathrm{SP}} - l_{g,n,\min}^{\mathrm{IP}}) \tag{28}$$
$$L_{g,n+1,\max}^{\mathrm{SP}} = L_{g,n,\max}^{\mathrm{SP}} + \lambda^{\mathrm{IP}}\cdot(o_{g,n}^{\mathrm{IP}} - L_{g,n,\max}^{\mathrm{SP}}). \tag{29}$$

In case of complete information (both sides know the cost function of the other stakeholder), the proposed algorithm yields the NBS, as computed in Section IV, in the first iteration. The final cost is the same as in the NBS:

$$\begin{aligned} \kappa_{\mathrm{IP},g}^{\mathrm{SP}} &= \frac{1}{2}\cdot(o_{g,n}^{\mathrm{SP}}+o_{g,n}^{\mathrm{IP}}) \\ &= \frac{1}{2}\cdot\left(\frac{2}{3}\cdot\left(L_g^{\mathrm{SP}}+\frac{1}{2}l_g^{\mathrm{IP}}\right)+\frac{2}{3}\cdot\left(l_g^{\mathrm{IP}}+\frac{1}{2}L_g^{\mathrm{SP}}\right)\right) \\ &= \frac{1}{2}\left(l_g^{\mathrm{IP}}+L_g^{\mathrm{SP}}\right) = \kappa_{\mathrm{IP},g,NBS}^{\mathrm{SP}}. \end{aligned} \tag{30}$$

However, IP and SP typically do not exchange their cost functions and private valuations. This incomplete information can lead to inefficiencies compared to the NBS with complete information [34]. For every cell $g \in G$, the resulting κ_g^{SP} and κ_g^{IP} may not result in the minimum cost for both stakeholders, but if both agree on a cost, it will be lower than $\tilde{\kappa}_g^{\mathrm{SP}}$ and $\tilde{\kappa}_g^{\mathrm{IP}}$ would be. This entire procedure is repeated for every cell $g \in G$, resulting in κ^{SP} and κ^{IP}, respectively.

D. Complexity Analysis

To discuss the complexity of the proposed approach, the update parameters have to be discussed first, since they play a crucial role with respect to how long the bargaining will take. The selection of the maximum iteration count N and the adjustment rates $\lambda^{\{\mathrm{IP,SP}\}}$ of the SP's and IP's offer are interdependent. A small adjustment rate $\lambda^{\{\mathrm{IP,SP}\}}$ leads to a high number of iterations until SP and IP either agree or finally disagree by repeating a previous offer. Choosing N too small will lead to more cells without an agreement, although it might be profitable for both to reach an agreement. A high value of $\lambda^{\{\mathrm{IP,SP}\}}$ is undesirable for the SP and IP, respectively, since the offer is corrected too much in favor of the other stakeholder.

We assume that both stakeholders agree on a maximum number of iterations, e.g., $N = 10$, due to time constraints, i.e., the bargaining should not take too much time compared to the duration of service placement. Each stakeholder can tune its adjustment rate $\lambda^{\{\mathrm{IP,SP}\}}$ individually, e.g., based on heuristics.

The bargaining procedure results in one of the two cases in each cell g: (i) the stakeholders agree on a cost or (ii) they disagree. In both cases, the number of iterations is bounded by N, since the bargaining is stopped after N rounds without agreement. The iterative bargaining is repeated for every cell $g \in G$. Thus, our iterative bargaining approach will converge at the latest after $N \cdot |G|$ iterations in the worst case. In the case of complete information, iterative bargaining yields the NBS, thus both stakeholders will agree on the payment stated in Eq. (30) for a particular cloudlet during the first iteration, which yields $1 \cdot |G|$ iterations in the best case.

VI. Experimental Evaluation

A. Case Study and Data Set

As discussed in Section I, cloudlets are especially suitable for processing latency-sensitive applications like mobile AR games. Therefore, our evaluation uses a real-world data set that we collected from Niantic's AR game Ingress, which was empirically investigated by Felka *et al.* [31] and is available on request for scientific purposes. In this game, players visit so-called portals that are linked to real points of interest and try to capture them, leading to continuous player movements in the real world. The data set contains the user's service requests including the location of a request, which will be called game *actions* in the following. One game action is one interaction between a player and a portal by either trying to capture or defend it.

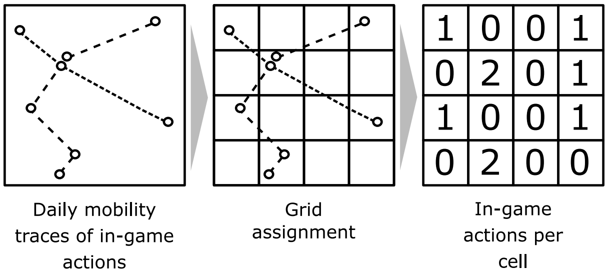


Fig. 3. Data preprocessing for every day.

TABLE III
MATHEMATICAL NOTATION AND THEIR VALUES FOR THE TAKE-IT-OR-LEAVE-IT APPROACH

Variable	Description	Value	Source
$\beta^{\mathrm{SP}}_{\mathrm{IP},g}$	Cost factor for processing unit SP has to pay to IP in cell g	$unif[\beta^{\mathrm{IP}}_g, 4\cdot\beta^{\mathrm{IP}}_g]$	[26], [28], [29]
$\phi^{\mathrm{SP}}_{\mathrm{IP},g}$	Cost for placing service in cell g SP has to pay to IP	$unif[\phi^{\mathrm{IP}}_g, 4\cdot\phi^{\mathrm{IP}}_g]$	[26], [28], [29]

The data set was collected over a time period starting on 1st of January 2016 until the 31st of May 2017 (i.e., almost 1.5 years), but we focus on the data collected for the year 2016. The data for this period is almost complete, except for a 5-day maintenance period (from 2016/09/20 to 2016/09/24). In the considered period of time, about $21,250$ users of the game have made over $17\cdot 10^6$ game actions at $53,259$ portals in the observation area. The data set covers urban, metropolitan, and rural areas with a total area of $25\,200\ \mathrm{k}^2\mathrm{m}$. We divided the area into cells of $1\,\mathrm{km}\times 1\,\mathrm{km}$ each, since 5G base stations approximately have a radius of about $500\ \mathrm{m}$, depending on the selected frequency and antenna. Subsequently, we assigned the game actions to the corresponding cells of the grid and summed all game actions on a daily basis within a cell to determine the total daily number of service requests of a cell. This process is visualized in Fig. 3. In our case study, the SP deploys the parts of the game (i.e., the service) relevant for the spatial area of the grid to cloudlets, while the IP owns the infrastructure, represented by the $25,200$ cells. In our scenario, each cell represents a possible deployment point for the services. The data set contains the number of in-game actions for each cell, representing the geographical service usage of Ingress.

B. Experimental Setup

Besides our iterative bargaining approach with two alternatives (PIS and NIS), we also include two additional cost models: a traditional Take-it-or-Leave-it approach and the NBS.

1) Take-It-or-Leave-It: The Take-it-or-Leave-it (ToL) approach represents a typical cost-based model. Here, the IP independently selects the prices in the cost functions in the following manner: First, placing a service in cell g incurs a fixed cost $\phi^{\mathrm{SP}}_{\mathrm{IP},g}\geq 0$ that the SP has to pay to the IP. This is the cost of the IP for placing the service on the cloudlet, i.e., transferring the required data from the cloud to the particular cloudlet or maintaining a service on a cloudlet. This implies that if in a previous bargaining the service was already placed on the cloudlet in cell g, ϕ^{IP}_g and thus $\phi^{\mathrm{SP}}_{\mathrm{IP},g}$ is only a fraction of the original cost, since the deployment cost is no longer included (the composition of ϕ^{IP}_g is explained at the end of Section VI-B on page 1833). Hence, the fixed cost $\phi^{\mathrm{SP}}_{\mathrm{IP},g}$ depends on the specific cell g. Second, the SP has to pay a cost factor $\beta^{\mathrm{SP}}_{\mathrm{IP},g}\geq 0$ to the IP for each of the p_g processing units required on the cloudlet in cell g, resulting in:

$$\kappa^{\mathrm{SP}}_{\mathrm{IP},g}=\phi^{\mathrm{SP}}_{\mathrm{IP},g}+\beta^{\mathrm{SP}}_{\mathrm{IP},g}p_g \tag{31}$$

TABLE IV
EVALUATION PARAMETERS

Variable	Description	Value	Source
$actions_g$	Number of game actions in cell g	Taken from data set	[31]
$\tilde{d}_g$	Data from BS to cloud if $\gamma_g=1$	$actions_g\cdot DU$	[35]
d_g	Data from BS to cloud if $\gamma_g=0$	$\tilde{d}_g\cdot(1-0.92)$	[19]
DU	Data units	$d\cdot\mathrm{MB}, d\in[1,8]$	[35]
p_g	Processing of actions	$actions_g\cdot PU$	[28], [29]
$\tilde{p}_g$	Processing in cloud even if $\gamma_g=1$	$p_g\cdot 0.1$	[19]
PU	Processing time	$[0.125\mathrm{s}, 2.5\mathrm{s}, 8.3\mathrm{s}]$	[36]–[38]
$\alpha^{\mathrm{SP}}_{\mathrm{CP},g}$	Cost factor for data SP has to pay to cloud provider in cell g	0.00012	[28], [29]
α^{IP}_g	Cost factor for data incurred for IP in cell g	0.00006	[26]
$\beta^{\mathrm{SP}}_{\mathrm{CP},g}$	Cost factor for processing SP has to pay to cloud provider in cell g	0.000029	[28], [29]
β^{IP}_g	Cost factor for processing incurred for IP in cell g	0.000087	[26]
ϕ^{IP}_g	Cost for placing cloudlet in cell g incurred for the IP	$128\,\mathrm{MB}\cdot\alpha^{\mathrm{SP}}_{\mathrm{CP},g}$	[26], [28], [29]
N	Number of bargaining iterations	10	Parameter discussion in Section V-D
λ^{SP}	Adj. rate of the SP's offer during iter. bargaining	$[0.1, 0.2, 0.3]$	
λ^{IP}	Adj. rate of the IP's offer during iter. bargaining	$[0.1, 0.2, 0.3]$	
$\hat{\alpha}^{\mathrm{IP}}_g$	Predicted cost factors from public sources	$unif[0.9,1.1]*\alpha^{\mathrm{IP}}_g$	Cost factor prediction in Section V-B1
$\hat{\alpha}^{\mathrm{SP}}_{\mathrm{CP},g}$		$unif[0.9,1.1]*\alpha^{\mathrm{SP}}_{\mathrm{CP},g}$	
$\hat{\beta}^{\mathrm{IP}}_g$		$unif[0.9,1.1]*\beta^{\mathrm{IP}}_g$	
$\hat{\beta}^{\mathrm{SP}}_{\mathrm{CP},g}$		$unif[0.9,1.1]*\beta^{\mathrm{SP}}_{\mathrm{CP},g}$	
$\hat{\phi}^{\mathrm{IP}}_g$		$unif[0.9,1.1]*\phi^{\mathrm{IP}}_g$	
$l^{\mathrm{IP}}_{g,min}$	SP's prediction of the IP's lower limit	$\hat{\phi}^{\mathrm{IP}}_g+\hat{\alpha}^{\mathrm{IP}}_g(d_g-\tilde{d}_g)+\hat{\beta}^{\mathrm{IP}}_g p_g$	
$L^{\mathrm{SP}}_{g,max}$	IP's prediction of the SP's lower limit	$\hat{\alpha}^{\mathrm{SP}}_{\mathrm{CP},g}(\tilde{d}_g-d_g)+\hat{\beta}^{\mathrm{SP}}_{\mathrm{CP},g}(p_g-\tilde{p}_g)$	

Table III summarizes the variables required for the ToL approach for this evaluation.

To place a service, the SP has to pay the price the IP asks for. The SP exclusively makes the placement decision. Only if the SP can reduce its cost by using a cloudlet, the SP performs a placement in the specific cell. This approach represents a lower-bound scenario in our evaluation, since the two stakeholders do not cooperate in any respect to achieve joint benefits.

2) Nash Bargaining: The second approach represents the optimal solution, i.e., the case where both stakeholders cooperate, have complete information about each other's cost, and achieve minimal social cost according to Section IV.

3) Iterative Bargaining: The third approach includes the two variants of our iterative bargaining approach. In the first variant, both stakeholders do not exchange any information. In contrast, in the second variant, information is partly shared between the stakeholders, e.g., the SP provides service usage predictions to the IP (see Section V).

The data set consists of in-game actions that reflect the service usage of the game but are not technical measures that we can directly use in our simulation. We therefore need to translate the in-game actions into technical measures such as the network traffic or the required processing power in a cell to conduct a meaningful simulation. In the following, we describe this process in more detail and present an overview of the evaluated parameters in Table IV.

Regarding network traffic, one hour of play in Ingress causes about 28 MB of traffic on a mobile device of one player, while a player performs an average of 14.56 game actions per hour. By dividing the network traffic of one-hour gameplay by the average number of game actions per hour, we get a value of 2 MB per game action. While 2 MB per game action seems like a lot, it includes all unrecorded activities, which also generate traffic (view the map or photos of the portals) and can thus generate traffic without performing any logged game action. Therefore, we consider an average of 2 MB per game action to be realistic. However, to avoid being specific for the game Ingress, we vary this parameter in our evaluation to match the network traffic of other popular AR games [35]. Thus, we assume that mobile AR games can generate data units DU ranging from 1 MB to 8 MB. If there is no placement of a service in cell g, we assume that the entire data has to be sent to the cloud, resulting in $\tilde{d}_g = actions_g \cdot DU$. Based on the results of a study by Wang *et al.* [3], we assume an average reduction of 92% in network traffic to the cloud and a remaining overhead traffic of 8%, if there is a service placement on a cloudlet in a particular cell, thus $d_g = \tilde{d}_g \cdot (1 - 0.92)$.

As already explained for the data units, we also vary the processing units to represent AR games with different processing requirements, which is based on required CPU cycles. Based on the works of Al-Shuwaili and Simeone [36], Liu *et al.* [37] and Dinh *et al.* [38], applications processed on cloudlets or similar edge computing resources require between $3 \cdot 10^8$ CPU cycles for simple applications and $20 \cdot 10^{10}$ CPU cycles for long-running applications. Based on the performance of modern CPUs, we assume an average of 2.5 GHz processing speed, resulting in 0.125 s and 8.3 s execution time, respectively. As a third service type we introduce an execution time of 2.5 s, resulting in $PU = [0.125, 2.5, 8.3]$. For processing one game action in the cloud (i.e., no cloudlet is used), we define $p_g = actions_g \cdot PU$. Furthermore, according to Wang *et al.*, $8\% - 12\%$ of requests need to be forwarded from the edge to the cloud server [3]. Therefore, we assume that 10% of all game actions require one additional processing unit in the cloud (e.g., to synchronize the game state), resulting in $\tilde{p}_g = p_g \cdot 0.1$.

As discussed in Section V-D, each stakeholder can tune its adjustment rate $\lambda^{\{\mathrm{IP},\mathrm{SP}\}}$ individually. Therefore, we select three values for $\lambda^{\{\mathrm{IP},\mathrm{SP}\}}$ and simulate all combinations of these three values between both stakeholders.

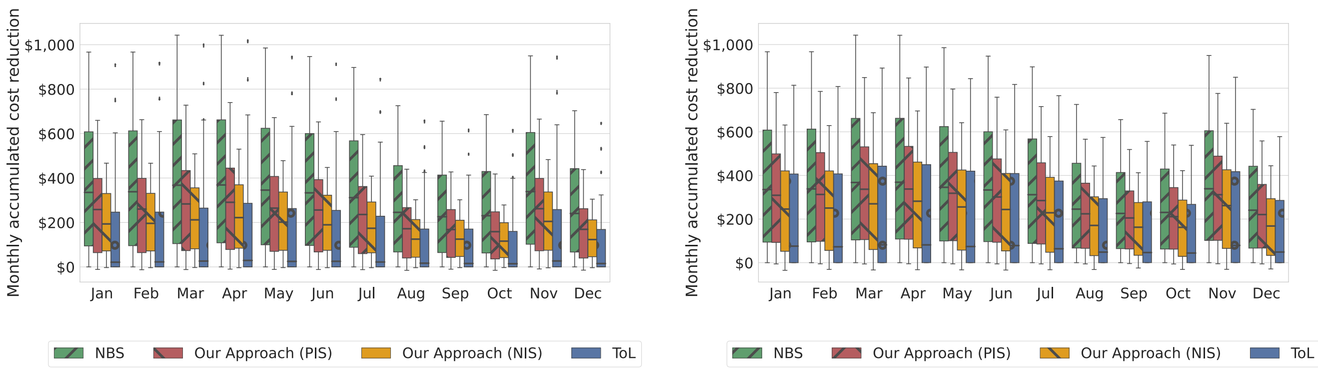
Now that the various service types have been defined, the following result emerges with regard to the number of simulations: for each combination of PU, DU, λ^{SP} and λ^{IP}, we simulate the four different bargaining approaches. Each simulation run simulates the placement behavior on a daily basis. This results in 311,040 simulations:

$$\begin{aligned} &|\text{Placement approaches}| \cdot |PU| \cdot |DU| \\ &\quad \cdot |\lambda^{\mathrm{IP}}| \cdot |\lambda^{\mathrm{SP}}| \cdot |days| \\ &\quad = 4 \cdot 3 \cdot 8 \cdot 3 \cdot 3 \cdot 360 = 311,040 \end{aligned} \tag{32}$$

Finally, we select and specify the parameters for the cost functions for the simulation. For data transmission cost factor to the cloud ($\alpha^{\mathrm{SP}}_{\mathrm{CP},g}$), we use public price lists of Amazon AWS Lambda [28] and Google Cloud Functions [29] and derive the cost the SP has to pay to the cloud provider per MB. Furthermore, based on Amazon Lambda@Edge [26], we derived that data transmission to edge resources (i.e., α^{IP}_g) costs about half of the data transmission to the cloud. Using the same approach, i.e., consulting public price lists of Amazon AWS and Google Cloud, we also could derive that one second of execution in the cloud costs about 0.000029 cents, while processing on edge resources is about three times more expensive. This results in $\beta^{\mathrm{SP}}_{\mathrm{CP},g} = 0.000029$ and $\beta^{\mathrm{IP}}_g = 0.000087$.

Accuracy in predicting service usage and accuracy in predicting the bargaining partner's cost factors characterize incomplete information. We assume that both stakeholders can predict each cost factor of the corresponding bargaining partner, in an interval of $\pm 10\%$ around the true value (see the predicted cost factors from public sources in Table IV). The accuracy of the prediction of service usage is assumed to be $\sigma^{\mathrm{p,SP}}_g = 0.5\sigma^{\mathrm{cell}}_g$, $\sigma^{\mathrm{d,SP}}_g = 0.5\sigma^{\mathrm{cell}}_g$ for the SP and $\sigma^{\mathrm{p,IP}}_g = 1.5\sigma^{\mathrm{cell}}_g$, $\sigma^{\mathrm{d,IP}}_g = 1.5\sigma^{\mathrm{cell}}_g$ for the IP, whereas σ^{cell}_g is a cell specific standard deviation given by the dataset. Therefore, the accuracy of service usage prediction varies between different cells.

To evaluate the ToL approach, the IP has to assign a cost factor the SP has to pay for processing on cloudlets (i.e., $\beta^{\mathrm{SP}}_{\mathrm{IP},g}$ and the initial placement of a service (i.e., $\phi^{\mathrm{SP}}_{\mathrm{IP},g}$). As indicated above, data transmission to the cloud is two times more expensive than data transmission to the cloudlet. Therefore, we assign the cost factor $\beta^{\mathrm{SP}}_{\mathrm{IP},g}$ as a random value between β^{IP}_g and $4 \cdot \beta^{\mathrm{IP}}_g$, which corresponds to twice the price on the average. We use the same approach to predict a value for $\phi^{\mathrm{SP}}_{\mathrm{IP},g}$



(a) Service provider

(b) Infrastructure provider

Fig. 4. Monthly accumulated cost reduction over one year for all approaches.

as $unif[\phi_g^{\mathrm{IP}}, 4 \cdot \phi_g^{\mathrm{IP}}]$, where ϕ_g^{IP} is defined as 128 MB $\cdot\, \alpha_{\mathrm{CP},g}^{\mathrm{SP}}$ because 128 MB is the smallest amount that can be used on AWS Lambda@Edge or Google Cloud Functions. Finally, maintenance cost is usually considered to amount to about 10% of the deployment cost [29], [39], which we also use as a reference in our evaluation.

C. Results

To evaluate the performance of our approach, we analyze the cost reduction, number of cloudlets used, and percentage of game actions processed on cloudlets. Since the cost reductions can vary in size between both stakeholders, i.e., distributed unequally between the two stakeholders, one stakeholder could benefit more than the other. Therefore, we also evaluate and compare the fairness of the approaches in a final step.

Since placing a service on a cloudlet incurs a fix cost ϕ_g^{IP} for the IP and $\phi_{\mathrm{IP},g}^{\mathrm{SP}}$ for the SP, there is a trade-off between the number of service placements and the number of game actions covered within that time period. Therefore, we simulate every single day by calculating every approach's placement decisions and the associated cost, network traffic, etc. This means that the stakeholders bargain every day. Since our iterative bargaining approach with partial information sharing (PIS) requires the SP to predict the service usage in advance, we add a slight variance based on a half daily standard deviation to the data in our data set to simulate inaccuracies in SP's prediction. In the case of iterative bargaining with no sharing of information (NIS), the IP performs the prediction. However, since the IP does not have internal information about the service, the IP can only roughly predict service usage. Therefore, the IP's prediction is not as good as the SP's prediction. To simulate the inaccuracies of the IP's prediction, we add one and a half standard deviation variance to the data to simulate the IP's prediction, which is an increase of one standard deviation compared to the SP's prediction of the PIS variant.

1) Cost Reduction: Fig. 4 shows the accumulated cost reductions of all approaches for each stakeholder over one year, where each box shows the accumulated cost reductions at the end of each month. The term "accumulated cost reduction" refers to the amount of money that IP and SP are saving (in our case per month) when using cloudlets compared to a purely cloud-based environment. In essence, it means

$$\text{Accumulated cost reduction} = \sum_{g \in G} \left[\tilde{\kappa}_g^{\{\mathrm{SP},\mathrm{IP}\}} \right] - \kappa^{\{\mathrm{SP},\mathrm{IP}\}} \tag{33}$$

Starting with the SP's perspective, the results show that the lower and upper quartiles of the ToL approach (blue boxes) are significantly lower compared to both variants of our iterative bargaining approach (red boxes for PIS, yellow boxes for NIS). However, rare cases exist where the ToL approach performs better than our iterative bargaining (e.g., if the service usage predictions for the iterative approach deviate significantly from the real values). On the average, both iterative bargaining variants lead to higher cost reductions. Furthermore, our PIS variant is quite close to the NBS. From the IP's perspective, the results are comparable. Similar to the previous perspective, our NIS and PIS bargaining approach perform better than the ToL approach and get close to the NBS solution. The mean savings are slightly lower for the PIS variant than for the NBS, but the variance of the cost reduction is larger for the NBS. Furthermore, in some rare cases both graphs show that the PIS and NIS approaches can also achieve negative cost reductions. In other words, both approaches may cause additional cost and do not reduce cost. The explanation for this effect is that both approaches perform their bargaining with the predicted service usage. In some rare cases, the predicted service usage may differ from the actual value in a way that the placement becomes unprofitable, causing additional cost. More specifically, if such a case occurred and additional monthly costs were generated, they ranged, on the average, from \$1.96 (NIS) to \$6.01 (PIS) for the SP and from \$2.88 (PIS) to \$17.01 (NIS) for the IP. In considering the average monthly savings of both approaches, the comparatively low cost that could occur in such an unusual case are rather marginal and negligible.

On the average, the SP achieves about 12% higher cost reduction with the NIS variant than with the ToL approach, while the IP achieves about 16% higher cost reduction if the IP also chooses the NIS variant of our iterative bargaining approach over the ToL approach. When considering the mean values of the PIS variant, the SP achieves about 44% higher cost reduction compared to the ToL approach, while the IP reaches about 47% higher cost reduction if the IP also

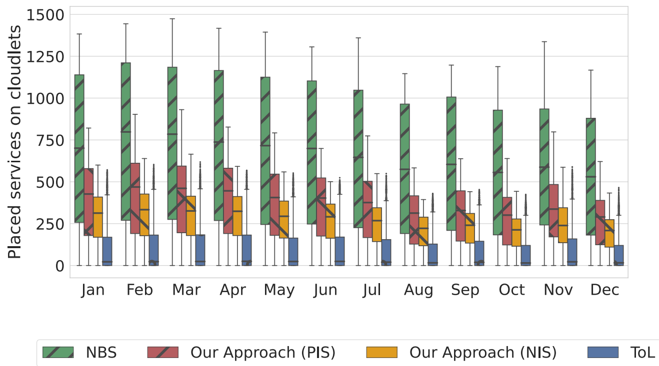


Fig. 5. The number of placed services on cloudlets for all approaches.

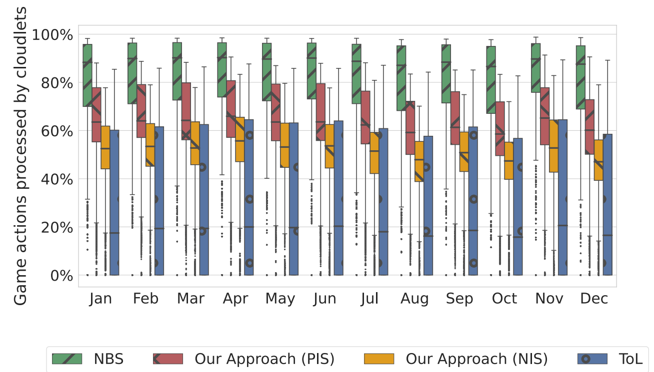


Fig. 6. Percentage of game actions processed on cloudlets for all approaches.

chooses the PIS variant of our iterative bargaining approach over the ToL approach. The differences in cost reduction are statistically significant (unpaired T-test, $p < 0.01$).

The cost reductions of the NBS (green boxes) are clearly higher than the cost reductions of the ToL approach, as shown by the blue boxes. Using the NBS, the IP can reduce the cost by an additional 78% compared to the ToL approach. The SP can reduce its cost up to about 117% compared to the ToL approach. Since both stakeholders share their cost reductions in the NBS, the cost reductions are identical for both. However, comparing the results of iterative bargaining with the results of the NBS solution, our NIS variant achieves 52% of the possible cost reductions for the SP and 65% of the possible cost reductions for the IP. In contrast, our PIS variant achieves 66% of the possible cost reductions for the SP and 83% of the possible cost reductions for the IP.

All in all, our approach performs better than the ToL cost-based approach in terms of cost reduction and comes close to the optimal solution under complete information.

2) Number of Placed Services: Fig. 5 shows the number of services placed on cloudlets for all four approaches over one year on a daily basis, grouped by month. The NBS represents the best case with an average number of 624.5 placed services on cloudlets, the ToL approach only achieves about 15% of this result with 94.9 placed services. Using our proposed approach in the NIS variant results in an average number of 248.5 placed services, 39.8% of the best case, leading to 2.6 more placed services compared to the ToL approach. The PIS variant results in an average number of 339.6 placed services, which is 54% of the best case, leading to 3.6 times more placed services compared to the ToL approach. The gaps between the NBS solution and PIS or NIS approaches appear to be quite significant. However, it is worth mentioning that the NBS approach has a 100% prediction accuracy, which results in a large number of placements that are on the verge of profitability. However, considering the number of placed services alone is less meaningful, since a high number of placed services does not necessarily lead to an improvement in the network, e.g., in case of a bad placement. Therefore, in the following we consider the average number of game actions performed by the placed services.

3) Game Actions Processed by Cloudlets: Fig. 6 shows the ratio of game actions processed directly by cloudlets to the total number of game actions. The number of game actions processed by cloudlets can be considered as an indicator for better QoE, since users experience a lower latency. With an average of 94.9 placed services, the ToL approach covers an average of 30% of all game actions, i.e., cloudlets compute 30% of all game actions.

While the NBS approach places 6.6 times more services on cloudlets, only 2.5 times more actions are covered (or 76% of all game actions). This is because service usage is not evenly distributed, with a few cells having high utilization. The majority of them have medium to low numbers of game actions. Therefore, a 6.6 times increase in cloudlets does not necessarily cover 6.6 times more actions. At this point, it is important to note that 76% coverage of game actions of the NBS approach is the optimal solution, where the joint savings are the highest. A coverage of 100% would also be theoretically possible. However, in our case study, it would not make sense economically, since additional placements would have to take place, causing additional cost and reduce the achieved savings. Both variants of our approach achieve between 48% (NIS) and 58% (PIS) of the covered game actions, while the ToL approach only achieves about 30%.

The figure also clearly shows that the lower quartiles of the ToL approach are significantly lower and close to zero compared to other approaches. The reason for this is that in the ToL approach, there are cells where a placement is only advantageous for the IP, e.g., if the price of the IP is too high. However, since the SP decides on the placement, it does not happen since the SP has no advantage. Our approach attempts to reduce cost across all stakeholders and achieves significantly better results than the ToL approach in terms of placed cloudlets and processed game actions.

Overall, the results show that cooperation is beneficial for both stakeholders. The IP can reduce the load in its infrastructure to reduce cost. If the IP also shares its cost reductions cooperatively with the SP, the SP deploys more services, leading to further cost reductions for the IP. The SP also benefits from the cooperative behavior by reducing the cost in the cloud. In total, the number of service placements on cloudlets is higher, which leads to more actions covered by cloudlets and a better quality of service, which in turn is beneficial for the customers of the IP and the SP.

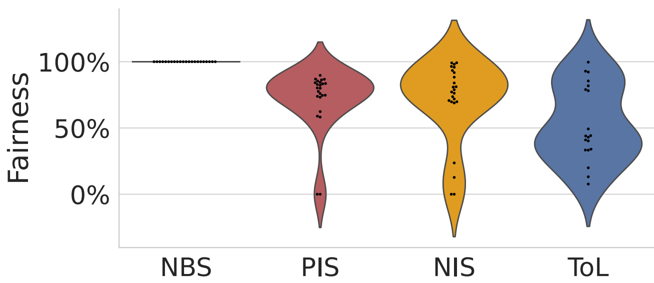


Fig. 7. Measured fairness of all approaches.

However, full cooperation between SP and IP is rather unrealistic, because it requires a full exchange of information between both stakeholders. Our novel iterative bargaining approach requires a minimum level of cooperation (e.g., exchange of predicted service usage) and yields a solution that comes close to a cooperation with complete information, without requiring the stakeholders to disclose their cost functions or further business-relevant internal details.

4) Fairness Comparison: If the IP shares its cost reductions cooperatively with the SP, the SP is able to deploy more services and further reduces the cost for the IP. This reflects a certain fairness from which both stakeholders can benefit. More precisely, the ratio between the savings of the IP and the SP reflects the fairness of the approaches. In the ideal case, both parties can achieve equal savings, resulting in a high fairness. To measure fairness, we use the well-recognized max-min fairness [40]. It is calculated of each approach using the accumulated savings once at the end of the simulation.

Fig. 7 contains a violin plot showing the distribution of the fairness for each approach. Starting with the NBS, the savings are identical for both stakeholders, resulting in a perfect fairness of 100%. Comparing the fairness of the ToL approach with the fairness of both variants of the presented approach, the fairness of the ToL approach is substantially lower and often below 50%. The reason for this is that in the ToL approach, the IP sets the price, which allows the IP to achieve a relatively high savings, as opposed to the SP. This leads to less fairness between both stakeholders. The distribution of fairness of the NIS and PIS variants are similar. As described in the previous section, the average accumulated savings differ between the two approaches. However, the ratio of accumulated savings between both stakeholders is balanced, which leads to a high fairness of both variants, with some cases, e.g., due to wrong service usage predictions, where the fairness can drop below 50%.

5) Impact of Incomplete Information: The results presented in the previous subsections show that different aspects of incomplete information yield different outcomes in terms of cost savings per stakeholder and fairness of the results. In the evaluation, we showed three different amounts of shared information. Using the NBS, both stakeholders have to share all information they have, whereas using the NIS variant of our approach no information is shared at all. Finally, in our approach using the PIS variant only the SP has to share little information about the service usage, which does not introduce any negative consequences, since the real service usage can be monitored by the IP anyway. These degrees of incomplete information are also reflected in the results. The accumulated cost reductions of SP (Fig. 4a) and IP (Fig. 4b) indicate that the more information is shared between the stakeholders, the more cost reductions both can achieve, resulting in NIS giving the least cost reductions and NBS the most, although even the NIS variant achieves better results compared to the traditional ToL approach. This is also visible in the number of placed services (Fig. 5) and game actions processed by cloudlets (Fig. 6), since these parameters directly influence the cost reductions. The different degrees of information sharing also impact the fairness of the bargaining (Fig. 7). The NBS, i.e., complete information sharing, always yields an absolutely fair outcome. The less information is shared, however, the more unfair the results will become. In summary, however, the bargaining approach with the lowest level of fairness and with the least cost reduction is still better with respect to all evaluated parameters than the ToL approach, where also no information is shared, but where additionally both stakeholders are not bargaining. This leads to the conclusion that stakeholders should cooperate and share at least a minimum amount of information. This way, they achieve near-optimal results in terms of cost savings, but without having to fully share all information, including private information, which is also reflected in the presented results for the PIS approach. Finally, the PIS approach seems more desirable as it yields better results across all evaluated parameters, but it must be taken into account that in the PIS approach the IP has to trust the SP. The SP could lie about the service usage prediction in its own favor. In scenarios where the SP is not trusted, the NIS variant offers an alternative. Furthermore, it is also possible to extend the PIS variant by a mechanism to enforce truthfulness, such as reputation-based methods [32].

VII. Conclusion

In this paper, we proposed a novel iterative bargaining approach between SP and IP for nearly optimal service placement in edge computing scenarios with respect to social cost despite incomplete information. We first introduced a cost model for SP and IP with relevant cost parameters, such as hardware deployment, service placement, processing, and data transfer. Assuming that both stakeholders share full information with each other, we derived the NBS as an optimal solution for both stakeholders, since social cost can be minimized. Since in practice it is unlikely that both stakeholders will share all of their information, we presented an iterative bargaining approach with two variants: (i) no information has to be shared at all, and (ii) minimum information is shared between IP and SP. Our approach finds a nearly optimal solution of about 83% of the NBS's cost reduction by sharing only service usage predictions and up to 65% of the optimal solution if no information is shared. Our case study based on the mobile AR game Ingress showed that despite incomplete information, our approach can achieve up to two times higher game action coverage on cloudlets than a traditional ToL cost model, which increases the users' QoE. Finally, we also showed how different degrees of incomplete information affect the outcome of our bargaining approach.

There are several areas of future work. For example, the cloud provider could be modeled as an active stakeholder in our game. Furthermore, the resource constraints of cloudlets should be considered. Finally, the proposed approach should be extended by a mechanism to enforce truthfulness between the stakeholders, such as reputation-based methods [32].

REFERENCES

[1] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

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

[4] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, Oct. 2009.

[5] J. Gedeon *et al.*, "From cell towers to smart street lamps: Placing cloudlets on existing urban infrastructures," in *Proc. IEEE/ACM Symp. Edge Comput. (SEC)*, Oct. 2018, pp. 187–202.

[6] Y.-Y. Shih, C.-Y. Wang, and A.-C. Pang, "Fog computing service provision using bargaining solutions," *IEEE Trans. Services Comput.*, vol. 14, no. 6, pp. 1765–1780, Nov. 2021.

[7] D. T. Nguyen, L. B. Le, and V. Bhargava, "Price-based resource allocation for edge computing: A market equilibrium approach," *IEEE Trans. Cloud Comput.*, vol. 9, no. 1, pp. 302–317, Jan. 2021.

[8] L. Chen, C. Shen, P. Zhou, and J. Xu, "Collaborative service placement for edge computing in dense small cell networks," *IEEE Trans. Mobile Comput.*, vol. 20, no. 2, pp. 377–390, Feb. 2021.

[9] Y. Liang *et al.*, "Interaction-oriented service entity placement in edge computing," *IEEE Trans. Mobile Comput.*, vol. 20, no. 3, pp. 1064–1075, Mar. 2021.

[10] Y. Zhang, L. Jiao, J. Yan, and X. Lin, "Dynamic service placement for virtual reality group gaming on mobile edge cloudlets," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 8, pp. 1881–1897, Aug. 2019.

[11] L. Liu and Q. Fan, "Resource allocation optimization based on mixed integer linear programming in the multi-cloudlet environment," *IEEE Access*, vol. 6, pp. 24533–24542, 2018.

[12] S. Mondal, G. Das, and E. Wong, "CCOMPASSION: A hybrid cloudlet placement framework over passive optical access networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2018, pp. 216–224.

[13] C. Ren, X. Lyu, W. Ni, H. Tian, and R. P. Liu, "Profitable cooperative region for distributed online edge caching," *IEEE Trans. Commun.*, vol. 67, no. 7, pp. 4696–4708, Jul. 2019.

[14] S. Yang, F. Li, M. Shen, X. Chen, and X. Fu, "Cloudlet placement and task allocation in mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 5853–5863, Jun. 2019.

[15] M. Jia, J. Cao, and W. Liang, "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," *IEEE Trans. Cloud Comput.*, vol. 5, no. 4, pp. 725–737, Oct./Dec. 2017.

[16] A. Mukherjee, D. De, and D. G. Roy, "A power and latency aware cloudlet selection strategy for multi-cloudlet environment," *IEEE Trans. Cloud Comput.*, vol. 7, no. 1, pp. 141–154, Jan./Mar. 2019.

[17] K. Peng, X. Qian, B. Zhao, K. Zhang, and Y. Liu, "A new cloudlet placement method based on affinity propagation for cyber-physical-social systems in wireless metropolitan area networks," *IEEE Access*, vol. 8, pp. 34313–34325, 2020.

[18] X. Cao, G. Tang, D. Guo, Y. Li, and W. Zhang, "Edge federation: Towards an integrated service provisioning model," *IEEE/ACM Trans. Netw.*, vol. 28, no. 3, pp. 1116–1129, Jun. 2020.

[19] T. Ouyang, R. Li, X. Chen, Z. Zhou, and X. Tang, "Adaptive user-managed service placement for mobile edge computing: An online learning approach," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2019, pp. 1468–1476.

[20] L. Chen, J. Xu, S. Ren, and P. Zhou, "Spatio–temporal edge service placement: A bandit learning approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 12, pp. 8388–8401, Dec. 2018.

[21] G. Dan and N. Carlsson, "Dynamic content allocation for cloud-assisted service of periodic workloads," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2014, pp. 853–861.

[22] Z. Yang, B. Y. Zhao, Y. Xing, S. Ding, F. Xiao, and Y. Dai, "AmazingStore: Available, low-cost online storage service using cloudlets," in *Proc. IEEE Int. Parallel Distrib. Process. Symp.*, vol. 10, Apr. 2010, p. 2.

[23] J. F. Nash, Jr., "The bargaining problem," *Econometrica, J. Econ. Soc.*, vol. 18, pp. 155–162, Apr. 1950.

[24] L. Gao, G. Iosifidis, J. Huang, L. Tassiulas, and D. Li, "Bargaining-based mobile data offloading," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1114–1125, Jun. 2014.

[25] J. Lahteenmaki, H. Hammainen, N. Zhang, and M. Swan, "Cost modeling of a network service provider cloud platform," in *Proc. IEEE Int. Conf. Cloud Eng. Workshop (IC2EW)*, Apr. 2016, pp. 148–153.

[26] Amazon.Com, Inc. (2021). *Amazon AWS Lambda Edge*. [Online]. Available: https://aws.amazon.com/lambda/edge/

[27] A. Mehta, W. Tarneberg, C. Klein, J. Tordsson, M. Kihl, and E. Elmroth, "How beneficial are intermediate layer data centers in mobile edge networks?" in *Proc. IEEE 1st Int. Workshops Found. Appl. Self Syst. (FASW)*, Sep. 2016, pp. 222–229.

[28] Amazon.Com, Inc. (2021). *Amazon AWS Lambda*. [Online]. Available: https://aws.amazon.com/lambda/

[29] Google. (Jul. 2021). *Cloud Functions | Google Cloud*. [Online]. Available: https://cloud.google.com/functions?hl=en

[30] Y. Li, H. Liu, W. Yang, D. Hu, X. Wang, and W. Xu, "Predicting inter-data-center network traffic using elephant flow and sublink information," *IEEE Trans. Netw. Service Manage.*, vol. 13, no. 4, pp. 782–792, Dec. 2016.

[31] P. Felka, A. Sterz, K. Keller, B. Freisleben, and O. Hinz, "The context matters: Predicting the number of in-game actions using traces of mobile augmented reality games," in *Proc. 17th Int. Conf. Mobile Ubiquitous Multimedia*, Nov. 2018, pp. 25–35.

[32] Y. Yan, J. Huang, X. Zhong, M. Zhao, and J. Wang, "Sequential bargaining in cooperative spectrum sharing: Incomplete information with reputation effect," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Dec. 2011, pp. 1–5.

[33] R. B. Myerson and M. A. Satterthwaite, "Efficient mechanisms for bilateral trading," *J. Econ. Theory*, vol. 29, no. 2, pp. 265–281, Apr. 1983.

[34] F. Farokhi and I. Shames, "Bilateral trade under information asymmetry and quantized measurements," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 5832–5837.

[35] (Sep. 2020). *Augmented Reality Games: How Much do They Really Cost*. [Online]. Available: https://ting.com/blog/augmented-reality-games/

[36] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Commun. Lett.*, vol. 6, no. 3, pp. 398–401, Jun. 2017.

[37] C.-F. Liu, M. Bennis, and H. V. Poor, "Latency and reliability-aware task offloading and resource allocation for mobile edge computing," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2017, pp. 1–7.

[38] T. Q. Dinh, Q. D. La, T. Q. S. Quek, and H. Shin, "Learning for computation offloading in mobile edge computing," *IEEE Trans. Commun.*, vol. 66, no. 12, pp. 6353–6367, Dec. 2018.

[39] Y. Li, W. Liang, and J. Li, "Profit maximization for service placement and request assignment in edge computing via deep reinforcement learning," in *Proc. 24th Int. ACM Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Nov. 2021, pp. 51–55.

[40] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1992.

Artur Sterz is currently pursuing the Ph.D. degree with the Department of Mathematics and Computer Science, Philipps-Universität Marburg, Germany. His current research focuses on wireless communication in mobile networks, 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).

Patrick Felka is currently 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.

Bernd Simon (Graduate Student Member, IEEE) received the B.Sc. and M.Sc. degrees in electrical engineering and information technology from the Karlsruhe Institute of Technology, Germany, in 2015 and 2019, respectively. He is currently pursuing the Ph.D. degree in electrical engineering with Technische Universität Darmstadt, Germany. He is 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 current research interest includes the application of game theory in wireless networks.

Sabrina Klos (née Müller) (Member, IEEE) received the B.Sc. and M.Sc. degrees in mathematics from Technische Universität Darmstadt, Germany, in 2012 and 2014, respectively, and the Ph.D. degree in electrical engineering from the Communications Engineering Laboratory in 2019. Her research interests include machine learning and optimization methods and their applications to wireless networks.

Anja Klein (Member, IEEE) is currently a Full Professor at Technische Universität Darmstadt, Germany, where she heading the Communications Engineering Laboratory. Her main research interests are in mobile radio technologies, including interference management, cross-layer design, relaying and multi-hop, computation offloading, smart caching, and energy harvesting. She has authored over 300 peer-reviewed papers and has contributed to 12 books. She is an inventor and a 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. She is a member of Verband Deutscher Elektrotechniker—Informationstechnische Gesellschaft (VDE-ITG).

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*, *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.

Bernd Freisleben (Member, IEEE) received the M.Sc. degree in computer science from Pennsylvania State University, USA, in 1981, and the Ph.D. and Habilitation degrees in computer science from the Technical University of Darmstadt, Germany, in 1985 and 1993, respectively. He is currently a Full Professor of computer science at the Philipps University of Marburg, Germany. He has published over 530 papers in peer-reviewed international journals and conferences, and he lead over 40 research projects funded by governmental institutions and industrial partners. His current research interests include distributed systems, mobile computing, and networked applications.