FOUNDATIONS OF COMPUTING AND DECISION SCIENCES
Vol. 44 (2019) No. 4
ISSN 0867-6356
DOI: 10.2478/fcds-2019-0020 e-ISSN 2300-3405

Cloud Brokering with Bundles: Multi-objective Optimization of Services Selection

Jedrzej Musial *, Emmanuel Kieffer, Mateusz Guzek, Gregoire Danoy, Shyam S. Wagle, Pascal Bouvry, Jacek Blazewicz † ‡

Abstract.

Cloud computing has become one of the major computing paradigms. Not only the number of offered cloud services has grown exponentially but also many different providers compete and propose very similar services. This situation should eventually be beneficial for the customers, but considering that these services slightly differ functionally and non-functionally -wise (e.g., performance, reliability, security), consumers may be confused and unable to make an optimal choice. The emergence of cloud service brokers addresses these issues. A broker gathers information about services from providers and about the needs and requirements of the customers, with the final goal of finding the best match.

In this paper, we formalize and study a novel problem that arises in the area of cloud brokering. In its simplest form, brokering is a trivial assignment problem, but in more complex and realistic cases this does not longer hold. The novelty of the presented problem lies in considering services which can be sold in bundles. Bundling is a common business practice, in which a set of services is sold together for the lower price than the sum of services' prices that are included in it. This work introduces a multi-criteria optimization problem which could help customers to determine the best IT solutions according to several criteria. The Cloud Brokering with Bundles (CBB)

*Corresponding author: Jedrzej.Musial@cs.put.poznan.pl

†M. Guzek, E. Kieffer, G. Danoy, Shyam S. Wagle and P. Bouvry: Computer Science and Communications Research Unit (CSC), University of Luxembourg, 6 avenue de la Fonte, L-4364 Esch-sur-Alzette, Luxembourg. G. Danoy, P. Bouvry: Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, 6 avenue de la Fonte, L-4364 Esch-sur-Alzette, Luxembourg. J. Musial and J. Blazewicz: Institute of Computing Science, Poznan University of Technology, ul. Piotrowo 2, 60-965 Poznan, Poland. J. Blazewicz: Institute of Bioorganic Chemistry, Polish Academy of Sciences, Z. Noskowskiego 12/14, 61-704 Poznan.

‡This study was partially supported by the Luxembourg National Research Fund (FNR, Luxembourg) and the National Centre for Research and Development (NCBR, Poland), through the IShOP project, INTER/POLLUX/13/6466384. E. Kieffer acknowledges the support of the Luxembourg National Research Fund (FNR, Luxembourg), with the AFR contract no. 9984150. The experiments presented in this paper were carried out using the HPC facility of the University of Luxembourg [40].

models the different IT packages (or bundles) found on the market while minimizing (maximizing) different criteria. A proof of complexity is given for the single-objective case and experiments have been conducted with a special case of two criteria: the first one being the cost and the second is artificially generated. We also designed and developed a benchmark generator, which is based on real data gathered from 19 cloud providers. The problem is solved using an exact optimizer relying on a dichotomic search method. The results show that the dichotomic search can be successfully applied for small instances corresponding to typical cloud-brokering use cases and returns results in terms of seconds. For larger problem instances, solving times are not prohibitive, and solutions could be obtained for large, corporate clients in terms of minutes.

Keywords: Applied operations research, Cloud computing, Cloud brokering, Optimization, Algorithms, Computational complexity

1. Introduction

Cloud computing [25] has become a widely-accepted paradigm allowing to access on-demand distributed computing and storage resources in a transparent and seamless way. It permits consumers to use a wide range of services without dealing with hardware or system configurations [3]. These benefits justify why the Infrastructure-as-a-Service (IaaS) model that offers a functional final application, is seen as the one with the highest potential in terms of market size [10]. New business models even feature free services (e.g., Google, Facebook) by relying on the commercialization of end-users data for marketing purpose.

In the past years the number of cloud services and cloud service providers has thus skyrocketed but with a certain lack of transparency. The plethora of available services provide companies the opportunity to establish robust IT solutions relying on security, reliability, redundancy and trust [28] while minimizing their overall cost. However, selecting the appropriate services at the best price is increasingly difficult for end-users. Indeed, numerous cloud services look-alike but in reality differ in terms of price, performance or reliability.

One of the answers to this problem is cloud service brokering [16], a model in which a trusted third party, the Cloud Service Broker (CSB) is matching the needs of users with services of providers. The involvement of CSBs is beneficial not only for customers, but also for providers, as CSBs can attract more customers by for instance suggesting the most appropriate services matching customers' needs.

Cloud service brokering faces many challenges. Among them there are the technical integration and customization of the various offers, the discovery and aggregation of the different non-uniform offers of providers, and the matchmaking between clients' needs and available offers. To complicate the picture, providers often apply bundling. A bundle consists of services from single provider and its price is lower than the price of the services that compose that bundle but sold separately. In other words, buying bundled products provides a discount. It will be proven in the paper that considering the notion of bundles makes the problem strongly-NP hard.

Despite the high relevance of the problem, to our best knowledge, it has not yet been formally modeled and studied in the literature. In this paper, we model the cloud brokering problem as multi-objective optimization one. To explore the problem, a dichotomic approach providing the exact solution set is applied. It ensures that all solutions on the convex hull of the Pareto front are found. This approach proves to be applicable on instances of reasonable size. Additionally, the set of exact solutions can be used as benchmark for approximation algorithms.

The main contributions of this paper are summarized as follows:

- The Cloud brokering with bundles (CBB) problem is defined.
- The proof of strong NP-hardness of CBB problem is provided.
- An instance generation tool is developed, based on the real data delivered by service providers.
- An exact algorithm solving the problem is presented. Extensive computational simulations highlight the specificity of the problem, that is the instance size solved in feasible time and the impact of the instance parameters.

The paper is organized as follows. Section 2 discusses the state of the art works related to the cloud brokering problem. Section 3 defines the cloud brokering problem and introduces the corresponding mathematical model. Section 4 includes the proof of the strong NP-hardness of the problem. Section 5 describes the benchmark generator based on the real data input. Section 6 presents the proposed multi-objective, exact solution method. Section 7 shows the experimental results accompanied by discussion. Section 8 summarizes the study and highlights the future work directions.

2. Related Works

Cloud brokering [25, 3] is one of the resource allocation problems crucial for cloud computing [15]. It can span across multiple layers of cloud computing, but some of them gathered more focus of researchers than others.

The International Organization for Standardization defines a cloud service broker (CSB) as a “cloud service partner that negotiates relationships between cloud service customers and cloud service providers” [17]. A cloud service partner is further explained as a “party which is engaged in support of, or auxiliary to, activities of either the cloud service provider or the cloud service customer or both”. In other words CSB becomes third party that negotiates relationship between (Cloud Service Providers, CSP) and (Cloud Service Client, CSC). It can change offers, modify them to fit customers’ needs and to fulfill providers demands at the same time.

Cloud brokers can be described and divided by their main functionalities (various specializations of cloud broker services). One can observe three main types of brokering services: aggregation, integration, and customization. Aggregation-type brokers provides platforms to bring together both providers and clients. The main idea is to presents offers from multiple CSP services at one central place - like a big CSP supermarket. This kind of platform is introduced as an unified service. In general it offers one billing (and provisioning) system. Client can order services from different providers in faster and more comfortable way. The most often aggregation CSBs provide wide variety of services and possibilities and serve different type of customers, from the individual ones to the biggest corporations. Integration CSB provide the new, common and unified system that is built on the basis of existing solutions. The main tasks to tackle are (but not limited to): integration of private and public clouds, bridging between CSPs, taking care of security issues, data exchange problems and security, proper sharing of data. Integrators are most often focused on the business-to-business sector. The customization type of brokers are supposed to be the most sophisticated from the three. On the one hand, they can include both aggregation and integration properties. On the other, they can offer other added-value services or even the new services that are dedicated (not offered by the CSPs) to the brokering system. Normally they are implemented in the CSB platforms, but in some cases they might include changes in the CSC's workflows. Customization CSBs are often specialized to offer unique purpose-oriented platforms (security, technology, specific field, local markets, managing, consolidation, etc.).

However, it is worth to note that in real-life cases, it's often impossible to clearly classify a company to the specific type of those three. Many of cloud brokers work in multiple fields, trying to respond to client needs.

Even in case of rational, perfectly-informed customers, brokering can provide advantages such as aggregating clients requests, better allocation of less flexible customers' demands, and balancing across providers infrastructures, as shown in [23], where a case study where brokering is compared with direct ordering of cloud services by customers was presented.

IaaS studies form the bulk of the literature on cloud brokering optimization, using well established models from scheduling, as well as distributed and parallel computing research. In this group of studies, the broker, typically a centralized entity, is allocating a set of user requests on cloud infrastructures offered by providers. These optimization problems may include the time dimension, becoming then scheduling problems. The most prominent objective is the cost minimization, but there could be additional objectives such as customer satisfaction, performance, or energy-efficiency related. Prasad et al. [32] proposed an algorithm (called CLOUD-CABOB) to solve the optimization problem where cloud users submit their requirements, and in turn vendors submit their offers containing price, QoS and their prepared sets of resources.

Nesmachnow, Iturriaga and Dorronsoro [30] propose a set of heuristics and a local search for allocation of users VM requests to instances reserved by a broker. Tordsson et al. [39] solve the VM placement brokering problem using integer programming provided by the CPLEX solver[1]. Nir et al. [31] also use the same integer programming approach and the CPLEX solver to optimize the brokering in case of cloud offloading, known also as hybrid cloud. Lucas-Simarro et al. [22] consider a practical problem of allocating VM instances on a set of clouds with OpenNebula [27] virtual infrastructure manager, choosing the cheapest allocation for each instance. Aazam and Huh [2]

[1]CPLEX is a product of IBM corp.

propose a dynamic broker, which predicts the behavior of user based on relinquish probability, that is the likelihood that the user will cease to use the requested services. The study involves also an advanced refund mechanism based on multiple criteria. It is further extended to Amazon cloud model and historical record integration [1]. Zhou et al. [46] propose a virtual resource renting methodology which maximizes the profit taking into account the priority of tasks and the estimations of future prices. Lucas-Simarro et al. [21] solve the IaaS problem in a dynamic pricing setting. Moens et al. study the problem of placement of application features on resources, including not only the server utilization cost, but also cost incurred by failed deployments of features [26]. The CLOUDRB framework [38] additionally extends the brokering model by adding energy-efficiency as an objective and extensive cloud job descriptions, which includes resources requirements and data transfer. The problem is solved by relying on particle swarm optimization. The cloud brokering IaaS problem may be also modeled as a negotiation process, solved by applying for instance an adaptive probabilistic behavioral learning system [33]. Kim et al. propose a brokering system for scientific workflows, which optimizes a multi-criteria problem using an aggregated objective function. The brokering part of the system selects the length of service period, to minimize the cost of VMs lease [19]. The idea of broker exploitation of pricing model is studied in [42] and solved using approximate dynamic programming. The theoretical study of user request aggregation under a concave cost function assumption together with Randomized Online Stack-Centric Scheduling Algorithm (ROSA) was proposed in [45].

SaaS brokering, despite its market importance, was less covered by the researchers. The SaaS aspect is covered as the parameters and arrival rate of tasks [9], however the problem is still modeled as an IaaS one.

Attempts to solve the problem in a distributed environment of multiple parties with diverging interest are commonly modeled as multi-agent problems. They commonly involve game theory to analyze the stable states of cloud brokering scenarios and optimal strategies for each party.

Gutierrez-Garcia and Sim [14] propose an agent based system, where a broker agent is responsible for composing of a complex services from simple services as requested by user agents. This work focuses on the feasibility of distributed composition rather than on the cost optimization. A work by Sim [37] describes an agent-based system with multiple competing brokers which maximizes the users utility functions defined in the terms of price and time-slot preferences. The problem is formulated as series of negotiations and is solved using a coordination scheme. Guan and Melodia [13] analyze the scenario with multiple cooperative brokers. They find out that cooperation is beneficial in settings with only few brokers, but its importance when there is large the number of brokers. Another study [34] shows that even for selfish service provider, long term cooperative strategy is more beneficial than naive maximization of each providers revenue.

The presented related works argues the validity and importance of the cloud services brokers in the cloud computing landscape. The game-theoretic part of the state of the art motivates additionally the usage of single broker, as it argues that cloud federations are more beneficial to cloud providers. Additionally, the literature focuses on resource-allocation motivates the advantages for users, who can benefit from aggregation and large-quantity discounts.

To the best of our knowledge, the problem of SaaS applications purchase is only covered by few studies. In general, there is no study that covers the aspect of the aspect of bundling.

It is worth mentioning that the Internet Shopping Optimization Problem (ISOP) [7] has some similarities with Cloud Brokering problem. Basic version of the ISOP could be defined as follows. A customer wants to buy a set of products $N = \{j = 1, \ldots, n\}$, where n is the number of products from online stores. A list of available shops is $M = \{i = 1, \ldots, m\}$, where m is the number of shops. One would like to buy all desired products at the minimum cost. This final cost also includes all delivery prices which are associated with the shops (the ones that one or more products were bought). The multiset N_i contains the available products from shop i. Each product $j \in N_i$ costs c_{ij}. Delivery cost is denoted as d_i for each shop i. The delivery cost is flat rate and it is charged just once if one or more products are bought from shop i. The ISOP is the minimization of the total cost of the shopping list N, including delivery costs. This is formally described as the finding of a disjoint selection of the products purchased from the different shops $X = (X_1, \ldots, X_m)$. Among all the products available at a given shop we should make a selection $X_i \subseteq N_i$. One have to buy all products from the shopping list, $\bigcup_{i=1}^{m} X_i = N$. The goal is to minimize total cost including the flat delivery costs: $F(X) = \sum_{i=1}^{m} \left(\delta \left(|X_i| \right) d_i + \sum_{j \in X_i} c_{ij} \right)$, where $|X_i|$ is the cardinality of the multiset X_i, and $\delta(x) = 0$ if $x = 0$ and $\delta(x) = 1$ if $x > 0$.

Few years ago Wojciechowski and Musial [43] described their idea for a web-based application / system dedicated to pharmacy products shopping. The idea was to propose different possibilities to customers to find shops in a geographically defined area that they can go and realize their shopping list at different total price. It could be perceived as a basic prototype of the ISOP, which was presented and formally modeled as an optimization problem [7]. It was proved that the problem is NP-hard in the strong sense. Seeing that the targeted application of ISOP should be an on-line program there was designed a simple, fast heuristic solution [44] proposed as a representative of greedy solutions. Beside its obvious advantages (very fast response, generally speaking good quality of results) subsequent analysis showed its solution inefficiency with a very specific (unrealistic, but still) situations and data sets. Therefore, a new Forecasting algorithm was proposed [5]. These heuristics are used to validate the correctness and performance of the proposed solution methods presented in the original article about ISOP [7].
Recently different types of the ISOP have been examined. Among other one should notice:

- version with price sensitive discounts [4, 5], where authors used an algorithm presented in [8],
- ISOP with two discounting functions (both based on the total amount of money spent in a shop) including price and shipping costs discounts [6]

Further evaluation of the Internet Shopping Optimization Problem with its optimization can be found in [20, 29]. Furthermore, some motivations and similarities can be found in research concerning stock market, namely considering multi-period portfolio optimization problem. Sawik introduced selected multi-objective methods for the problem where the objective is to allocate the wealth on different securities [35]. Goal is to optimize the portfolio expected return, the probability that the return is not less than a required level. Problem can be somehow similar with the CBB problem when we link stocks, securities with services and bundles.

Models and algorithms issued from ISOP appear useful for addressing the general Cloud Brokering problem. Cloud Brokering with Bundles definition and modeling also can benefit from the above mentioned knowledge and research.

3. Problem Definition

The Cloud Brokering with Bundles (CBB) is an optimization problem. To introduce the problem we decided to define and analyze the computational complexity of the single objective CBB problem. Subsequently, we presented the bi- and multi-objective versions of the problem, where the modeling and possible solutions are combined of many single objective problems. The objective (single version) of CBB is to minimize the cost of the selection of cloud services, which are sold in bundles, such that all cloud services required by one customer are included in chosen bundles. Additionally, each provider has a limited number of each type of the bundle. The problem is formally defined as (cf. Table 1 for the denotation of symbols):

$$\min \quad F = \sum_{i \in P} \sum_{j \in B^i} d_{ij} x_{ij}, \tag{1}$$

$$\text{s.t.} \quad \sum_{i \in P} \sum_{j \in B^i} q_{ij}^s x_{ij} \geq r^s \quad \forall s \in \{1, ..., |S|\}, \tag{2}$$

$$x_{ij} \leq u_{ij} \ \forall i \in \{1, ..., |P|\}, \ \forall j \in \{1, ..., |B^i|\}, \tag{3}$$

$$x_{ij} \in \mathbb{N}, \tag{4}$$

$$q_{ij}^s \in \mathbb{N}. \tag{5}$$

The objective function F (Eq. 1) is equal to the total cost of all bundles purchased from all providers, where x_{ij} is a non-negative integer, which determines how many bundles of type j are ordered from provider i. There are four constraints. The first constraint (Eq. 2) ensures that the number of obtained service units is greater or equal to the required number of service r^s, where s is the service type. The number of obtained service units is calculated as the sum of the numbers of ordered bundles multiplied by the number of service s units available in bundle j (q_{ij}^s).
The second constraint (Eq. 3) ensures that the number of bundle type j units bought from provider i (denoted as x_{ij}) is not higher than the maximum number of bundle type j units available at provider i (namely u_{ij}). According to cloud computing paradigm [36] a user should not need to take care of the limitations, but in practice

there are often limitations on how many service instances can be ordered ad hoc, even for the largest cloud providers.
The third and fourth constraints (Eqs. 4, 5) ensure that both x_{ij} and q^s_{ij} are natural numbers ($\mathbb{N}$).

Table 1. Table of symbols

Symbol	Explanation
F	Objective function
P	Set of providers
B	Set of bundle types
B^i	Bundle set of provider i
S	Set of service types
b_j	bundle of type j
d_{ij}	price associated with bundle type j sold by provider i
q^s_{ij}	number of units of service type s included in bundle j sold by provider i
r^s	required number of service units of type s
u_{ij}	maximum number of bundle type units j available at provider i
x_{ij}	number of bundle type j units bought from provider i (decision variable)

4. Computational Complexity

In this section we analyze the computational complexity of the Cloud Brokering with Bundles (CBB) problem. Moreover, we will prove its NP-hardness by proving the NP-completeness of its decision counterpart problem (let us call it CBB-D). The latter has the same input as CBB plus additional parameter y. The question is whether or not there exists a selection of bundles containing all of the required services with the total cost of y or less. Furthermore, we shortly discuss possibilities of CBB approximation.

Proposition 1. *Cloud Brokering with Bundles problem is strongly NP-hard even if all costs of bundles are equal to one and there is just one provider.*

Proof. Let us construct a pseudo-polynomial transformation from the well-known, strongly NP-complete problem, Set Covering (Minimum Cover) Problem (SCP) [18, 12] to the CBB-D problem.
The SCP can be described as follows. There is a given finite set of elements R, a collection C of subsets of R, and a positive integer $K \leq |C|$. The question is whether there is a possibility of covering the set R with K or less subsets $C' \subseteq C$ ($|C'| \leq K$)

such that every element of R belongs to at least one member of C'?
It is understandable that if Y is a solution to SCP, then $|Y| \leq K$.

Given an instance of SCP we construct the following instance of CBB-D. Assume that there is just one provider i, therefore $|P| = 1$ and we can skip index i in the following parts of the section. The provider offers j bundle types of services which create a set of all bundles $\bigcup_{j=1}^{|B|} b_j = B = C$. The customer would like to purchase s types of services, what creates set $S = \{s_1, s_2, \ldots, s_{|R|}\}$, $S = R$. Each service s should be included in at least one bundle $b_j \subseteq B$, $b_j = C'$. Required number of units of each service type s is 1 (therefore there is a requirement to collect all s service types from the set S), all prices d_j are equal to 1, number of service types s included in each bundle type j, sold by provider is $q_j^s = 1$, and the maximum number of bundle types j available at the provider (u_j) equals 1. The threshold value of the criteria is $y = K$.

Now, we show that SCP has a solution if and only if there exists a solution X for the constructed instance of problem CBB-D with $|X| \leq y$. Seeing that the transformation is pseudo-polynomial, problem CBB-D belongs to the class NP.

Let Y be a solution to SCP. Construct a solution for problem CBB-D, in which the required services are realized with K bundles determined by $C' = b_j \in Y$, i.e., $x_j = 1$ if $b_j \in Y$ and $x_j = 0$ if $b_j \notin Y$. Since Y is a covering of S, all the required services are purchased, and the cost of the corresponding solution X is $|X| \leq K$.

Now assume that there exists a solution X for problem CBB-D with the cost $|X| \leq y$. For this solution the total number of chosen bundle types $(\sum_{j=1}^{|B|} x_j)$ should not exceed y because otherwise $|X| > y$ and corresponding solution Y will be $|Y| > y$. On the other hand, the number of bundles should not be less than y because otherwise at least one service $s \in S$ will not be purchased. Therefore, there are exactly y bundles j with $x_j = 1$ $(\sum_{j=1}^{|B|} x_j = y)$. Since all purchased services covers the set S, the collections of chosen bundles (where $x_j = 1$) is $\bigcup_{j=1}^{|B|} x_j b_j = C'$ which represents a solution for SCP. □ □

Proposition 2. *By upholding the CBB assumptions of the Proposition 1 and following the literature [12] we can post that the problem is solvable in polynomial time (matching techniques) if all $b_j \in B$ have $|b_j| \leq 2$.*

It is worth mentioning that the SCP is considered as a very difficult to approximate [24]. Therefore, we can introduce the following statement.

Statement 1. *The CBB problem cannot be approximated within a factor of $c \cdot \log N$ in polynomial time, for any $0 < c < \frac{1}{2}$, unless $\mathcal{P} = \mathcal{NP}$.*

Since the single objective CBB is modeled and proven to be NP-hard there is a clear observation that the bi- and multi-objective versions of CBB problem that will be combined of many single-objective CBBs are at least as hard as the latter.

Table 2. Gamma distribution parameters fitted for real services

Shape	1.58	0.92	1.07	1.92	1.18	2.59	0.51	1.19	0.85	0.75	0.99
Rate	9.33	3.79	12	18.34	10.72	435.9	5.51	36.4	3.68	2.96	10.98

5. Bi-objective CBB and Instance Generation

Customers have generally several criteria when they purchase goods and services. Therefore, after a proof of NP-hardness for the single-objective version, we will directly follow with a bi-objective version of the problem for the experiments described hereafter. The first objective is naturally the cost which is desired to be at least as possible. The second objective could be any kind of indicator concerning security, reliability, trust. In our case, we will generate an artificial second objective and ensure that it is negatively correlated with first one.

$$\min \quad (F_1 = \sum_{i \in P} \sum_{j \in B^i} d^1_{ij} x_{ij}, F_2 = \sum_{i \in P} \sum_{j \in B^i} d^2_{ij} x_{ij}) \tag{6}$$

$$\text{s.t.} \quad \sum_{i \in P} \sum_{j \in B^i} q^s_{ij} x_{ij} \geq r^s \quad \forall s \in \{1, ..., |S|\} \tag{7}$$

$$x_{ij} \leq u_{ij} \ \forall i \in \{1, ..., |P|\}, \ \forall j \in \{1, ..., |B^i|\} \tag{8}$$

$$x_{ij} \in \mathbb{N} \tag{9}$$

$$q^s_{ij} \in \mathbb{N}. \tag{10}$$

In order to stay close to the Cloud Service Market, we gathered data from existing cloud providers. In total, 19 cloud providers were data have been gathered are: AWS, Centurylink Cloud, Microsoft Azure, Rackspace Cloud, HP Cloud, Elastic Host, Google Cloud, Upcloud, Vault Network, SoftLayer, BareMetal Cloud, Exoscale, Aruba Cloud, Cloudsigma, DigitalOcean, City Cloud, GMO Cloud, and Cloud Central. For each of them, the advertised price of proposed services were recorded. As each provider offers a unique subset of services, we decided to select the services for which the most complete data was available. It resulted in a set of 11 services, with some missing data points.

The gamma distribution was selected as appropriate for fitting the data, as the histograms show that the distribution is not symmetric and the density is highest close to the minimal price values, as presented in Fig.1. These results confirm an intuitive expectation: due to the competition, CSPs have to provide price as low as possible, but there still exist some providers that offer expensive alternatives, hence the significant long tail. Additionally, gamma distribution shape can vary greatly depending on parameters, which ensures its adaptability to different real-world situations. In the end it was possible to fit gamma distributions for 11 services with parameters shown in Table 2.

According the bi-objective model introduced in section 5, (q^s_{ij}) have been generated using a uniform distribution $\mathcal{U}(1, 10)$. The generation of the number of required

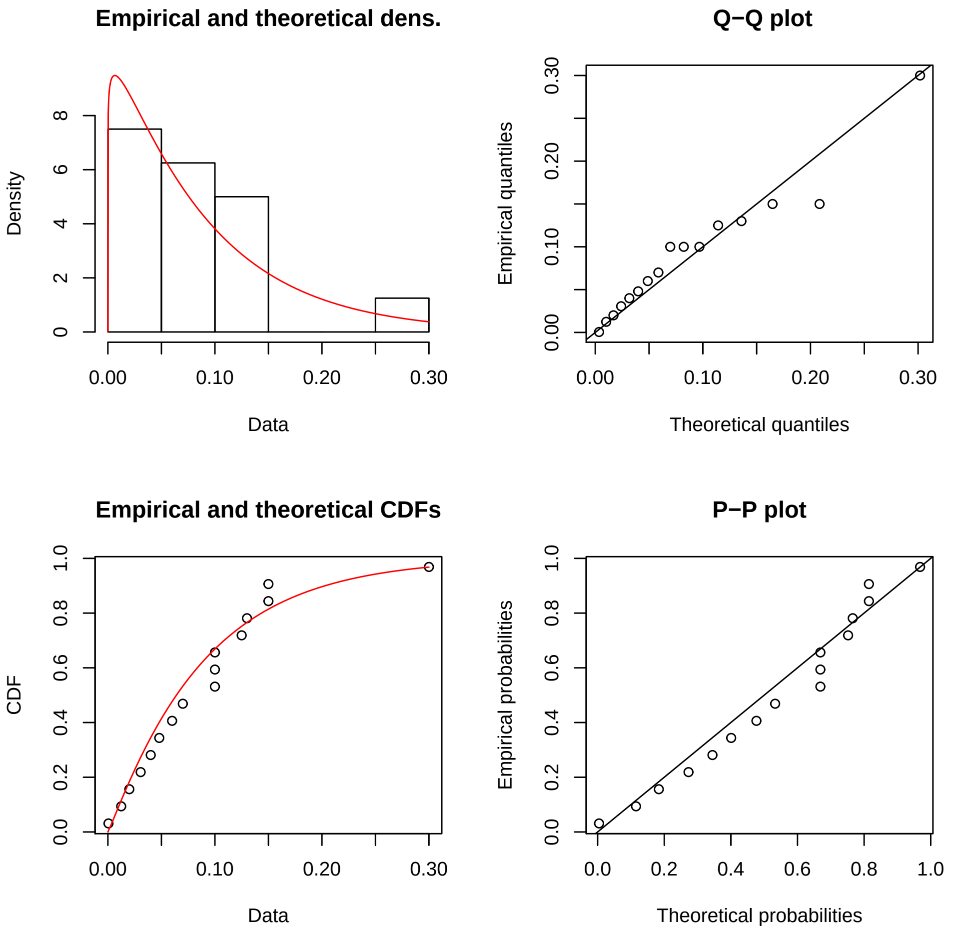


Figure 1. Example of the quality of gamma distribution fitting to a service pricing data (expressed in general units).

services r^s is chosen according to the uniform distribution $\mathcal{U}(10, 50)$. Bundle prices are generated according to the following expression:

$$d^1_{ij} = \mathcal{U}(\max_{s \in q_{ij}} cost_{is}, \sum_{s \in q_{ij}} cost_{is} \times q^s_{ij}) \tag{11}$$

where $cost_{is}$ is the price of service s for provider i. $cost_{is}$ are generated according to the computed gamma distribution. In summary, the price of a bundle lies between the price of the most expensive services and the sum of all service prices in the bundle. To ensure that all generated instances are feasible, we also consider *atomic bundles* which are single service bundles with guarantee that the demand can be satisfied. Finally, the second objective F_2 is generated with the same principle but kept negatively correlated with F_1 representing the prices of all bundles.

6. Algorithms Design

Cloud Brokering optimization often involves multiple objectives. In this paper, we decided to solve instances with two objectives using a dichotomic search approach. This procedure (see Algorithm 1) iteratively solves multi-objective problem by mapping it to a series of single objective problems. The single-objective problems are all weighted-sum versions of the multi-objective problem, but the weights are dynamically set to find new Pareto-optimal solutions [11]. It is only able to provide supported solutions and has been used as first phase procedure in [41]. The problem is not tractable, but it can be efficiently solved optimally for relatively large instances.

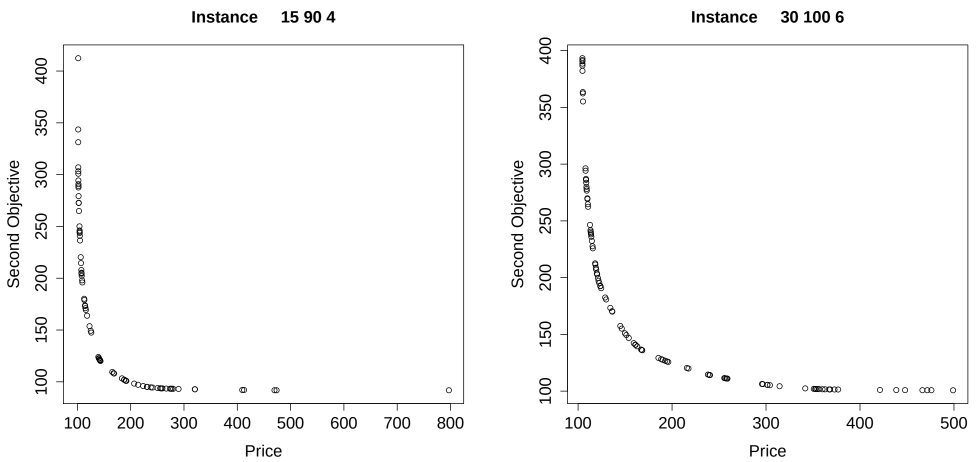
Algorithm 1 Dichotomic search

Compute $x^{(A)} = min\{\lambda_1 F_1(x) + \lambda_2 F_2(x) : x \in X\}$ with $\lambda_1 = 1$ and $\lambda_2 = 0$
Compute $x^{(B)} = min\{\lambda_1 F_1(x) + \lambda_2 F_2(x) : x \in X\}$ with $\lambda_1 = 0$ and $\lambda_2 = 1$
$S = \{x^{(A)}, x^{(B)}\}$
proc dichomotic($x^{(A)} \downarrow$,$x^{(B)} \downarrow$,$S \updownarrow$)
Compute $x^{(C)} = min\{\lambda_1 F_1(x) + \lambda_2 F_2(x) : x \in X\}$ with $\lambda_1 = F_2(x^{(A)}) - F_2(x^{(B)})$ and $\lambda_2 = F_1(x^{(B)}) - F_1(x^{(A)})$
if $x^{(C)} \notin S$ **then**
    $S = S \cup \{x^{(C)}\}$
    dichomotic($x^{(A)}$,$x^{(C)}$,S)
    dichomotic($x^{(C)}$,$x^{(B)}$,S)
end if

7. Experimental Evaluation

The proposed method is a weighted sum scalarization approach which discover all supported solutions of the optimal Pareto front. The experimental evaluation is therefore devoted to two main factors: time needed to solve instance depending on its size, and the number of solutions on the convex hull of the Pareto front, further referred to as the number of solutions. The size of instance is driven by three parameters: number of service types, the number of providers, and number of bundles per provider. Table 3 summarizes the values of parameters which were set to balance the range and precision of values with the time and resources needed to perform the simulations. We consider that the maximum size of instance, which includes 100 types of services, 30 providers and 8 non-trivial bundles offered by provider, is realistic even for large institutional customers or large groups of aggregated individual users, keeping in mind that each type of service is typically requested multiple time in a single instance.

For each combination of parameters, 30 instances were randomly generated and solved. Experiments have been conducted on the High Performance Computing (HPC) platform of the University of Luxembourg [40]. The IBM ILOG CPLEX 12.4 solver has been used on a single core of a machine with Intel Xeon E5-2660 CPU at 2.2GHz with 64Gb of RAM wit GNU/Linux Debian 7 operating system. For each run, the machine used for computations was fully reserved to this single task to limit variability. The maximum allowed solving time for each bi-objective instance of the problem was set to 4 hours. The total solving time corresponded to 54.4 days of single-core computation.



(a) (b)

Figure 2. Examples of convex hulls of Pareto fronts (prices expressed in general units)

Based on the usage of the standardized experimental platform, we can meaningfully compare the solving time of instances. The number of services and number of providers have the highest impact on the solution time, and as presented in Fig. 3b, these impacts are multiplicative. The number of bundles per provider is less signif-

Table 3. Parameters for Instance Generation

Parameter	Values
Service Types	10,20,30,40,50,60,70,80,90,100
Providers	5,10,15,20,25,30
Bundles per Provider	0,2,4,6,8

icant and is additive with other parameters (see Fig. 3a and Fig. 3c). The average solving time is up to over 2000 seconds, however in most simple cases it is below 500 s. It is clearly prohibitive to be offered for individual customers as an online web-service, but such customers are probably interested in the smallest instance sizes and an approximate Pareto front. On the other hand, this time is acceptable for institutions that make large purchases which involve significant budgets.

The solving time is clearly influenced by the number of solutions that compose the convex hull of the Pareto front. After analysis of the results, it can be observed that, contrary to the solving time, it is dependent almost entirely on the number of services, as illustrated in Fig. 4a and Fig. 4b. The two remaining parameters have little impact on the solution number (Fig. 4c). The average number found Pareto fronts is almost linear to the number of services. The fronts are rich in solutions, and are often clustered, as presented in Fig. 2. It leads to the conclusion that for the practical considerations it may be more efficient to solve the problem using only few predefined alternatives of weights, which should find representatives of limited number of clusters with high diversity.

Such approach would be then dependent on the average time spent to find a single solution from the Pareto front, as the number of optimization solving subprocedures would be fixed. Because of that, it is important to analyze the average time per single solution criterion, which combines the two already discussed evaluation criteria. This criterion is useful also in cases when the full Pareto front is not necessary, but only a single (or limited number of) solution(s) with predefined weights. It is important to note that this time can be also interpreted as the average time of solving single-objective version of the problem.

The results show that for extreme values of some parameters, the general trend does not apply. It is well visible for the low values of service number, where the effects of another parameter are either increased (as for bundles per provider, Fig. 5a) or decreased (as for providers number, Fig. 5b). There is also slight increase of the importance of bundles per provider in case of low number of providers, as presented in Fig. 5c. The vast majority of cases can be solved in few seconds, which we find acceptable for on-line service exposed to end customer. Such response time would also allow an interactive usage: similarly to search engines available for flights or hotels, customers could refine their search if the solution is not fitting their needs.

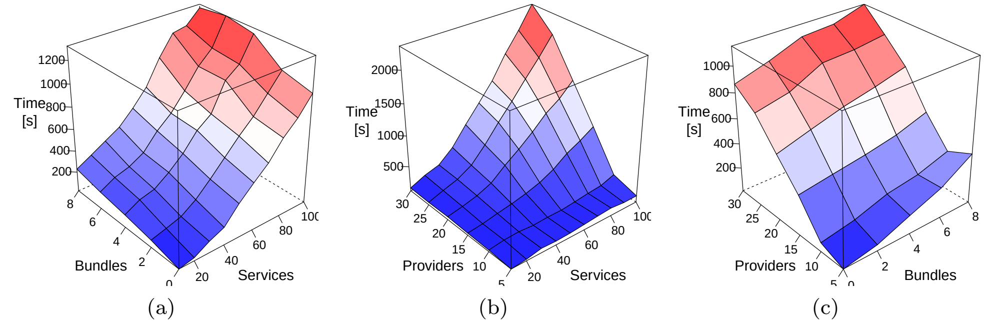


Figure 3. Average time needed by the dichotomic approach to find the convex hull of Pareto front

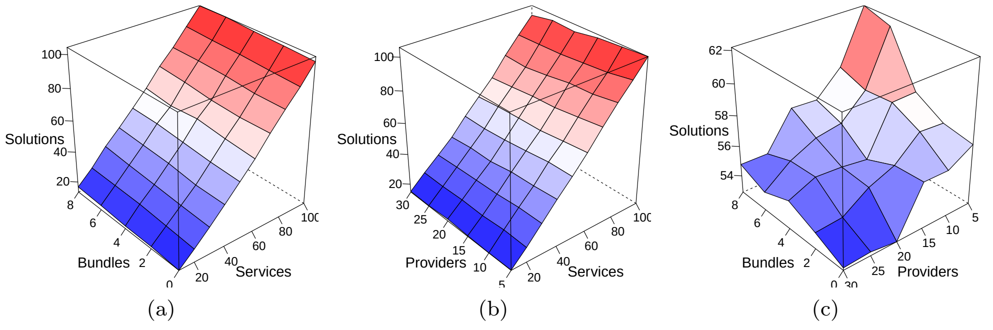


Figure 4. Average solution number on the convex hull of Pareto front

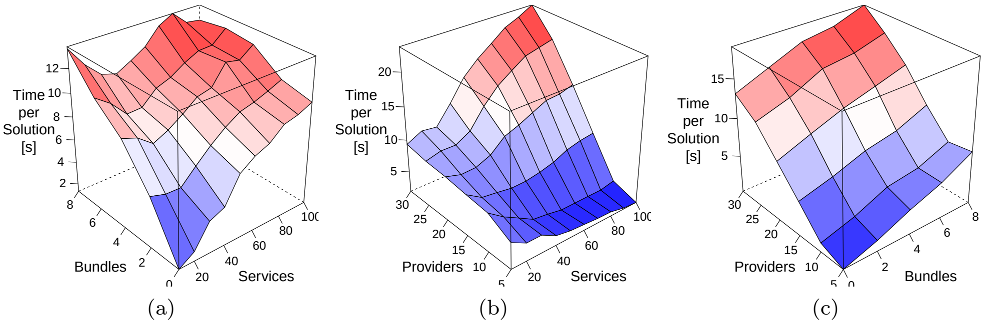


Figure 5. Average time to obtain a solution from the convex hull of Pareto front

8. Conclusions

In this paper we defined and model a new optimization problem that concerns services in Cloud Computing, the Cloud Brokering Bundle problem. The problem directly replies to the needs of users of such systems, however it was not investigated up to now. We prove that the problem is strongly NP-hard even in its single-objective variant. The most interesting and practical bi-objective version of the problem should therefore be even harder. We propose also an instance generator that is based on pricing data gathered from 19 real cloud providers.

The experiments revealed that number of services has the impacts on the number of solutions in the obtained Pareto front. The number of available providers also impacts on the time to solve, while the number of bundles is of lesser importance. The experimental investigation shows that the intended for typical use cases instances of the problem can be solved in acceptable time using exact methods.

This study introduces single- and bi-objective cases. In case of higher number of dimensions, the dichotomic approach may be prohibitively costly, as number of solutions may grow exponentially with each additional objective. Such multi-objective cases, which are definitely corresponding to the real-life need are part of future work. They could be solved by heuristics or computational intelligence tools such as evolutionary algorithms.

References

[1] Aazam M., Huh E., St-Hilaire M., Lung C., and Lambadaris I. Cloud Customer's Historical Record Based Resource Pricing. *IEEE Trans. Parallel Distrib. Syst.*, 27(7):1929–1940, 2015.

[2] Aazam M. and Huh E.-N. Cloud broker service-oriented resource management model. *Trans. Emerg. Telecommun. Technol.*, 28(2):e2937, 2017.

[3] Armbrust M., Fox A., Griffith R., Joseph A., Katz R., Konwinski A., Lee G., Patterson D., Rabkin A., Stoica I., and Zaharia M. A view of cloud computing. *Commun. ACM*, 53(4):50–58, 2010.

[4] Blazewicz J., Bouvry P., Kovalyov M. Y., and Musial J. Erratum to: Internet shopping with price-sensitive discounts. *4OR-Q J Oper Res*, 12(4):403–406, 2014.

[5] Blazewicz J., Bouvry P., Kovalyov M. Y., and Musial J. Internet shopping with price sensitive discounts. *4OR-Q J Oper Res*, 12(1):35–48, 2014.

[6] Blazewicz J., Cheriere N., Dutot P.-F., Musial J., and Trystram D. Novel dual discounting functions for the Internet shopping optimization problem: new algorithms. *J. Sched.*, 19(3):245–255, 2016.

[7] Blazewicz J., Kovalyov M. Y., Musial J., Urbanski A. P., and Wojciechowski A. Internet shopping optimization problem. *Int. J. Appl. Math. Comput. Sci.*, 20(2):385–390, 2010.

[8] Blazewicz J. and Musial J. E-Commerce Evaluation – Multi-Item Internet Shopping. Optimization and Heuristic Algorithms. In Hu B., Morasch K., Pickl S.,

and Siegle M., editors, *Operations Research Proceedings 2010: Selected Papers of the Annual International Conference of the German Operations Research Society*, pages 149–154. Springer, Berlin, Heidelberg, 2011.

[9] Calheiros R., Ranjan R., and Buyya R. Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments. In *Parallel Processing (ICPP), 2011 International Conference on*, pages 295–304, Sept 2011.

[10] Columbus L. Roundup Of Cloud Computing Forecasts And Market Estimates, 2016. www.forbes.com/sites/louiscolumbus/2016/03/13/roundup-of-cloud-computing-forecasts-and-market-estimates-2016, 2016. Accessed: 2016-06-16.

[11] Ehrgott M. *Multicriteria Optimization.* Springer-Verlag, Berlin Heidelberg, 2005.

[12] Garey M. R. and Johnson D. S. *Computers and Intractability, A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, New York, 1979.

[13] Guan Z. and Melodia T. The Value of Cooperation: Minimizing User Costs in Multi-broker Mobile Cloud Computing Networks. *IEEE Trans. Cloud Comput.*, PP(99):1–1, 2015.

[14] Gutierrez-Garcia J. O. and Sim K. M. Agent-based Cloud service composition. *Appl. Intell.*, 38(3):436–464, 2012.

[15] Guzek M., Bouvry P., and Talbi E.-G. A Survey of Evolutionary Computation for Resource Management of Processing in Cloud Computing [Review Article]. *IEEE Comput. Intell. Mag.*, 10(2):53–67, May 2015.

[16] Guzek M., Gniewek A., Bouvry P., Musial J., and Blazewicz J. Cloud Brokering: Current Practices and Upcoming Challenges. *IEEE Cloud Comput.*, 2(2):40–47, Mar 2015.

[17] International Telecommunication Union. Information technology — Cloud computing — Overview and vocabulary. Technical Report ITU-T Y.3500, International Organization for Standardization, 2014.

[18] Karp R. M. Reducibility among Combinatorial Problems. In Miller R. E., Thatcher J. W., and Bohlinger J. D., editors, *Complexity of Computer Computations*, The IBM Research Symposia Series, pages 85–103. Springer US, 1972.

[19] Kim S.-H., Kang D.-K., Kim W.-J., Chen M., and Youn C.-H. A Science Gateway Cloud With Cost-Adaptive VM Management for Computational Science and Applications. *IEEE Syst. J.*, 11(1):173–185, 2016.

[20] Lopez-Loces M. C., Musial J., Pecero J. E., Fraire-Huacuja H. J., Blazewicz J., and Bouvry P. Exact and heuristic approaches to solve the Internet shopping optimization problem with delivery costs. *Int. J. Appl. Math. Comput. Sci.*, 26(2):391–406, 2016.

[21] Lucas-Simarro J. L., Moreno-Vozmediano R., Montero R. S., and Llorente I. M. Scheduling strategies for optimal service deployment across multiple clouds. *Future Gener. Comput. Syst.*, 29(6):1431–1441, 2013.

[22] Lucas-Simarro J. L., Moreno-Vozmediano R., Montero R. S., and Llorente I. M. Cost optimization of virtual infrastructures in dynamic multi-cloud scenarios. *Concurr. Comput.: Pract. Exp.*, 27(9):2260–2277, 2015.

[23] Ludwig A. and Schmid S. Distributed Cloud Market: Who Benefits from Specification Flexibilities? *SIGMETRICS Perform. Eval. Rev.*, 43(3):38–41, Nov. 2015.

[24] Lund C. and Yannakakis M. On the hardness of approximating minimization problems. *J ACM*, 41(5):960–981, Sept. 1994.

[25] Mell P. and Grance T. The NIST definition of cloud computing. *Natl. Inst. Stand. Technol.*, 53(6):50, 2009.

[26] Moens H., Truyen E., Walraven S., Joosen W., Dhoedt B., and De Turck F. Cost-Effective Feature Placement of Customizable Multi-Tenant Applications in the Cloud. *J. Netw. Syst. Manag.*, 22(4):517–558, 2013.

[27] Moreno-Vozmediano R., Montero R. S., and Llorente I. M. IaaS Cloud Architecture: From Virtualized Datacenters to Federated Cloud Infrastructures. *IEEE Comput.*, 45(12):65–72, 2012.

[28] Musial J. and Lopez-Loces M. C. Trustworthy Online Shopping with Price Impact. *Found. Comput. Decis. Sci.*, 42(2):121–136, 2017.

[29] Musial J., Pecero J. E., Lopez-Loces M. C., Fraire-Huacuja H. J., Bouvry P., and Blazewicz J. Algorithms solving the Internet shopping optimization problem with price discounts. *Bull. Pol. Ac. Sci.: Tech. Sci.*, 64(3):505–516, 2016.

[30] Nesmachnow S., Iturriaga S., and Dorronsoro B. Efficient Heuristics for Profit Optimization of Virtual Cloud Brokers. *IEEE Comput. Intell. Mag.*, 10(1):33–43, Feb 2015.

[31] Nir M., Matrawy A., and St-Hilaire M. Economic and Energy Considerations for Resource Augmentation in Mobile Cloud Computing. *IEEE Trans. Cloud Comput.*, PP(99):1–1, 2015.

[32] Prasad G. V., Prasad A. S., and Rao S. A combinatorial auction mechanism for multiple resource procurement in cloud computing. *IEEE Transactions on Cloud Computing*, 6(4):904–914, 2018.

[33] Rajavel R. and Thangarathanam M. Adaptive Probabilistic Behavioural Learning System for the effective behavioural decision in cloud trading negotiation market. *Future Gener. Comput. Syst.*, 58:29–41, 2016.

[34] Samaan N. A Novel Economic Sharing Model in a Federation of Selfish Cloud Providers. *IEEE Trans. Parallel Distrib. Syst.*, 25(1):12–21, Jan 2014.

[35] Sawik B. Selected Multiobjective Methods for Multiperiod Portfolio Optimization by Mixed Integer Programming. In Lawrence K. D. and Kleinman G., editors, *Applications in Multicriteria Decision Making, Data Envelopment Analysis, and Finance*, volume 14 of *Applications of Management Science*, pages 3–34, Bingley, UK, 2010. Emerald Group Publishing Limited.

[36] Shawish A. and Salama M. *Cloud Computing: Paradigms and Technologies*, pages 39–67. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.

[37] Sim K. M. Agent-Based Cloud Computing. *IEEE Trans. Serv. Comput.*, 5(4):564–577, Fourth 2012.

[38] Somasundaram T. S. and Govindarajan K. CLOUDRB: A framework for scheduling and managing High-Performance Computing (HPC) applications in science cloud. *Future Gener. Comput. Syst.*, 34:47–65, 2014.

[39] Tordsson J., Montero R. S., Moreno-Vozmediano R., and Llorente I. M. Cloud brokering mechanisms for optimized placement of virtual machines across multiple providers. *Future Gener. Comput. Syst.*, 28(2):358–367, 2012.

[40] Varrette S., Bouvry P., Cartiaux H., and Georgatos F. Management of an academic HPC cluster: The UL experience. In *Proc. of the 2014 Intl. Conf. on High Performance Computing & Simulation (HPCS 2014)*, pages 959–967, Bologna, Italy, July 2014.

[41] Visée M., Teghem J., Pirlot M., and Ulungu E. Two-phases Method and Branch and Bound Procedures to Solve the Bi–objective Knapsack Problem. *J. Glob. Optim.*, 12(2):139–155, 1998.

[42] Wang W., Niu D., Liang B., and Li B. Dynamic Cloud Instance Acquisition via IaaS Cloud Brokerage. *IEEE Trans. Parallel Distrib. Syst.*, 26(6):1580–1593, June 2015.

[43] Wojciechowski A. and Musial J. A customer assistance system: Optimizing basket cost. *Found. Comput. Decis. Sci.*, 34(1):59–69, 2009.

[44] Wojciechowski A. and Musial J. Towards Optimal Multi-item Shopping Basket Management: Heuristic Approach. In Meersman R., Dillon T., and Herrero P., editors, *On the Move to Meaningful Internet Systems: OTM 2010 Workshops*, volume 6428 of *Lecture Notes in Computer Science*, pages 349–357, Berlin, 2010. Springer-Verlag.

[45] Zhang R., Wu K., Li M., and Wang J. Online Resource Scheduling Under Concave Pricing for Cloud Computing. *IEEE Trans. Parallel Distrib. Syst.*, 27(4):1131–1145, 2016.

[46] Zhou A., Sun Q., Sun L., Li J., and Yang F. Maximizing the profits of cloud service providers via dynamic virtual resource renting approach. *EURASIP J. Wirel. Commun. Netw.*, 2015(1):1–12, 2015.

Received 28.02.2019, Accepted 21.08.2019