PricingTheCloud: A Pricing Estimator for an Informed Cloud-Migration Process

Ines Raissa Djouela Kamgang[1][0000−0002−1564−1084], Abdessalam Elhabbash[2][0000−0002−9463−0446], and Yehia Elkhatib[3][0000−0003−4639−436X]

[1] Department of computer Engineering, Faculty of Engineering and Technology, University of Buea, Cameroon ines.djouela@ubuea.cm

[2] School of Computing and Communications, Lancaster University, UK a.elhabbash@lancaster.ac.uk

[3] School of Computing Science, University of Glasgow, UK yehia.elkhatib@glasgow.ac.uk

Abstract. One of the main challenges facing businesses migrating to the cloud is getting an estimate of their costs in advance. The estimators available to date allow companies to compare the different virtual machine offerings from each operator, but only venture very slightly into estimating the overall cost, which includes operational and network costs. Existing estimators include operational costs in their estimates, but almost no one considers the network cost, which is a complex but far from negligible component. In this paper, we seek to address this issue by proposing a new estimator called *PricingTheCloud*. It is an estimator that enables companies to have an accurate estimate of their costs in advance. Unlike other estimators, *PricingTheCloud* considers network costs in the cost estimation. Its evaluation shows an average accuracy of 86.73% for compute costs and 65.44% for network costs in different AWS-to-AWS scenarios as compared to AWS invoices and shows the effectiveness of the proposed estimator compared to three other cloud costs estimators namely, Cloudorado, Holori, and Vantage.

Keywords: Cloud computing · Cost calculation · Pricing models · AWS · Compute costs · Network costs · Infrastructure as a Service (IaaS)

1 Introduction

Cloud computing has grown at an unprecedented rate and has become the de facto choice for obtaining highly available computing resources. More and more organizations and businesses are accelerating their transition from self-hosted to cloud-hosted systems that have the advantages of being more cost-effective, secure, flexible, reliable, and sustainable [4]. However, a number of questions acutely arise: Which cloud service provider (CSP) should the company choose? What type of virtual machines (VMs) or instances should be selected? Which VM configuration will match the needs of the enterprise? Which VMs offer the most cost-effective solutions for a specific workload and usage patterns? This final question is the one we aim to answer here. However, the answer to this and similar questions can be quite complex for the three main reasons.

First, there are many CSPs on the market. Three CSPs lead the pack of public cloud services with 65% of the market share [10, 16], namely Amazon Web Services-AWS (33%), Microsoft Azure (22%), and Google Cloud Platform-GCP (10%). Other CSPs include Alibaba, Digital Ocean, IBM, and Oracle to name a few. *Second, the number and heterogeneity of Infrastructure as a Service (IaaS) instances offered by CSPs continually increase.* Each of the above-mentioned providers offers different types of cloud instances, each with its specific features and properties. As a result, the number and type of cloud instances developed and released each year are constantly growing [15]. For instance, the number of Linux-based instance types has almost doubled between 2015 and 2022 [8, 18]. *Third, providers offer various pricing strategies.* Every CSP has its own pricing strategy that involves a bespoke calculation of how VMs are being used and how much network traffic is being transferred.

Thus, making the best choice for a given customer means choosing the best CSP, the type of instances that suitably match their needs, and the regions and availability domain/zone (AD/AZ) where the selected instances will be deployed.

While CSPs publish their pricing structures and some even offer proprietary cost calculators, accurately estimating the total cost of cloud resources remains a complex task due to the intricate and often opaque specifics of each provider's pricing model. Notably, these CSP-backed cost calculation tools often present a significant user burden due to their extensive input requirements.

In addition to the proprietary tools offered by CSPs, industry-based estimators have been developed to address these issues. However, the main function of these estimators is to compare the offerings of different CSPs without providing an accurate estimate of the total cost of the cloud resource used. Those that do, provide an erroneous estimate of the total cost, as they consider the prices related to computing and storage resources but neglect those generated by the network traffic between entities. This cost is far from negligible and is a point of misunderstanding for many customers.

To address these concerns, this paper makes the following contributions:

1. A new flexible, transparent, and low-complexity estimator to help companies have a complete cost estimate of their systems, regardless of their architectural structure.
2. An accurate total cost computation model that considers not only compute, but also network parameters.
3. An accuracy assessment of the values generated by the proposed estimator as compared to ground truth values obtained from real AWS deployments.
4. An assessment of the efficiency of our generated results in comparison to three other estimators.

The remainder of this article is organized as follows. Section 2 summarizes relevant related work in cloud service provider estimation and costing. The cost function used to implement the estimator is presented in section 3. Section 4 details the different algorithms used in the implementation. Section 5 evaluates and validates the accuracy and the efficiency of the proposed estimator compared to real results generated by AWS invoices and other estimators. Section 6 details limitations and threats to validity and section 7 concludes the study.

2 Related Works

Many approaches have been proposed to handle the complexity of computing and estimating total cloud costs. Those approaches are divided into two main categories, which are industry-based approaches and research-based approaches. In this section, we present relevant existing work in each category.

2.1 Industry solutions

Most industry-based approaches rely on the development of estimator engines. From these, we focus on Helix BMC [3], Cloudorado [1], and Holori [2], highlighting their added values and shortcomings compared to our proposed approach.

The *Helix BMC* simulator helps companies in their migration process by estimating the monthly cost of replicating a given on-premise architecture in the cloud. The demo version, which is free of charge, allows the user to enter only three VMs in the system. However, it presents two main shortcomings. First, the simulator does not allow the user to choose the geographical location of the VM that the user wants whereas, in all the providers' offerings, the cost of a VM is highly dependent on its geographical location. As a result, the computational cost estimates are not realistic. Second, the total cost estimated at the end of the simulation process does not consider network costs since it does not allow the user to enter the amount of data transferred in and out of each VM.

Cloudorado is an estimator engine that compares IaaS cloud offerings between three CSPs: AWS, Azure, and GCP. The aim is to help companies make an informed migration process. The comparison occurs at three levels: server hosting, storage, and providers. However, even though the simulation interface seems more comprehensive than that of Helix BMC (e.g. it gives the possibility to enter network cost requirements such as inbound and outbound transfers), the estimator still presents a major shortcoming: it only compares offerings from different providers, but it does not give an estimation of the total cost

Finally, the *Holori* simulation platform functions similarly to Cloudorado but supports more than three CSPs. However, like Cloudorado, this estimator does not take network costs into account and does not calculate the total cost.

Overall, we observe that all industry estimators fall short in providing efficient total cost calculation engines that consider all strategic elements (compute, storage, and network costs), even within the architecture of a single cloud provider.

2.2 Academic efforts

Several studies have highlighted the lack of transparency and certainty in estimating the costs of cloud computing services [9, 17, 13, 11]. Research approaches to cloud cost estimations often rely on mathematical models. Kratzke [12] and Brumec et al. [6] present considerations for estimating cloud costs and express this mathematically. They propose algorithms for cost estimation including computing, storage, networking, and database costs. Aldossary et al. [5] present an energy-based cost model for VMs in the cloud, aiming to express the total cloud costs with a focus on energy savings. However, their cost estimation is limited to computing parameters and does not account for network cost parameters. Cho and Bahn [7] present a real-time cost estimation model for IaaS resources by monitoring resource usage, but leaves out network costs. It also assumes a linear relationship between resource usage and cost, which is not necessarily true.

For the aforementioned reasons, there is a need for an estimator that considers all relevant parameters, including compute and network costs, when estimating total costs, ultimately aiming to simplify the experience for cloud users.

3 Problem Formulation

To formally model the cloud cost estimation problem, we begin by defining the fundamental components of our cloud cost model, namely the operational compute and network costs, as well as the architecture on which they depend.

Let $G = (N, L)$ be a directed graph representing the architecture for which a user would like to estimate the costs after migration to the cloud. N represents the set of VMs in the architecture, and L the set of links between the VMs. Each VM is characterized by its CPU, memory, hourly rate, location, CSP, and processing time. The location consists of the triplet continent, region, and availability domain/zone.

The global expression of the total cost (TC) is given by equation 1, where $Total\_OP\text{Cost}$ is the total operational cost and Total_NCost is the total network cost.

$$TC = Total\_OP\text{Cost} + \text{Total\_NCost} \tag{1}$$

The total operational cost paid for hiring compute resources is expressed by:

$$Total\_OP\text{Cost} = \sum_{VM \in N} processTime(VM) * hourrate(VM, location) \tag{2}$$

where $processTime(VM)$ is the processing time of the considered VM and $hourrate$ its hourly rate. Note that a VM's hourly rate depends on its provider and location. For example, a VM launched in US-Ohio will not have the same price as one with the same characteristics launched in Asia Pacific-Tokyo.

The total network cost represents the cost for the volume of traffic exchanged between the VMs of a given architecture. This is expressed as follows:

$$\text{Total\_NCost} = \sum_{l \in L} dataRate(l) * cost\_per\_dataRate \tag{3}$$

where $dataRate(l)$ is the volume of traffic exchanged on a unidirectional link l separating two VMs of the graph, and $cost\_per\_dataRate$ is its data rate cost.

While Equation 3 presents a fundamental formula for calculating network cost, the actual computation process is considerably more nuanced. The cost per data rate associated with each link in the cloud architecture is not a static value; rather, it fluctuates based on a confluence of factors. These factors include the specific CSPs that host the VMs at each end of the link, as well as the geographical distribution of these VMs across continents, regions, and availability zones. Moreover, in scenarios where communication occurs between VMs hosted by different CSPs, the volume of traffic exchanged becomes an additional factor that influences the cost per data rate.

This intricate interplay of variables, combined with the inherent complexity of network pricing structures, has led to the omission of network costs in a substantial number of existing cloud cost estimation tools. This omission, however, is a significant oversight, given the substantial impact that network costs can have on the overall financial outlay associated with cloud infrastructure. In contrast, our proposed estimator seeks to bridge this gap by incorporating a meticulous and comprehensive analysis of network cost invoicing practices employed by CSPs.

4 *PricingTheCloud* Implementation

Leveraging the aforementioned pricing specifics, we have formulated a generalized cost model and developed the requisite algorithms for our cloud cost estimator. The core engine of the estimator comprises three primary algorithms: total operational cost computation, total network cost computation, and total cost aggregation. The design of these algorithms has been guided by a thorough analysis of the unique pricing characteristics associated with major cloud providers, as discussed previously. These algorithms are described in detail below.

Algorithm 1 computes the total operational cost. Its input is a list of all the desired VMs along with their specified parameters. These parameters include CPU, memory, execution time, and CSP. Furthermore, each VM is associated with a dictionary detailing all its linked VMs and their respective data rates. The initial step involves matching the user's desired VM to the VMs available in the CSP database (line 5). It is important to note that the AWS CSP database, at the time of extraction, contained over $8,000$ VM instance types across various regions, each with its distinct pricing characteristics. After filtering, the selected VM is added to a list of chosen VMs named *VMlist_ afterDbSearch*. The list of chosen VMs is then taken as the input of the operational function to generate for each VM of the new list its operational cost following equation **??** (line 9). The result is summed up and the total operational cost is returned (line 11).

Algorithm 2 calculates the total network cost. It takes as input the list of VMs of the previous algorithm. For each VM, it retrieves all connections, and determines for each connection to which provider the two constitutive endpoints belong. Thus, depending on whether the two endpoint VMs belong to the same

Algorithm 1: Total Operational Cost Computation Algorithm

Data:
VM_list : list of all the VMs of the desired architecture and their corresponding parameters;
$VMlist\_afterDbSearch$: list of all the VMs after the pre-processing stage;
Result:
Total_OPCost : The total operational cost;

Begin
Total_OPCost = 0
$VMlist\_afterDbSearch = []$
for $(VM \in VM\_list)$ **do**
    $chosen\_VM = findTheBestVm\_fromDb()$
    $VMlist\_afterDbSearch += chosen\_VM$
end
for $(VM \in VMlist\_afterDbSearch)$ **do**
    $VM\_OPCost = VM.hourrate * VM.processTime$
    Total_OPCost$+= VM_OPCost$
end
return Total_OPCost

provider or not, the function encompassing the corresponding logic is called for execution and a network cost for that specific case is generated. The function `getNCost_AWS_to_AWS()` (line 5) is called if the two endpoint VMs belong to AWS, and `getNCost_AWS_to_overseas()` (line 7) if only the egress VM resides in AWS. Finally, the network costs computed at each level are aggregated.

The algorithm that computes the total cost is quite simple and will not be explicitly represented. It takes as input the results of algorithms 1 and 2 and generates the total cost by summing up those results following equation 1.

5 Evaluation

To evaluate *PricingTheCloud*, we benchmark its compute and network cost estimates against ground truth values derived from AWS invoices and against state-of-the-art commercial estimators *Cloudorado* [1], *Holori* [2], and *Vantage* [19]. It is worth mentioning that to the best of our knowledge, no other research paper has developed such an estimator for us to use as a baseline.

5.1 Accuracy assessment against AWS invoices

A study we conducted allowed us to point out the fact that the cost of the assessed architecture also varies depending on the geographical position of the VMs in the architecture. Specifically, the network cost will then vary if the VMs are in the same region or not. The accuracy assessment needed to consider these potential variations to be efficient. For this purpose, our accuracy assessment is

Algorithm 2: Total Network Cost Computation Algorithm

```
Data:
VM_list : list of all the VMs of the graph and their corresponding
parameters;
link_list : list of all the links between VMs and their corresponding data rate;
VM_a: the VM initiating the traffic (egress VM)
VM_b: the VM terminating the traffic (ingress VM)
Result:
NC : The total network cost;
Begin
NC = 0
for (link ∈ link_list) do
    if vm_a.CSP = "AWS" and vm_b.CSP = "AWS" then
        NCost+ = getNCost_AWS_to_AWS();
    else if vm_a.CSP = "AWS" and vm_b.CSP != "AWS" then
        NCost+ = getNCost_AWS_to_overseas() ;
    else
        print ("No such provider in our database")
    end
end
return NCost
```

conducted following two different scenarios on the AWS platform. The first scenario evidences intra-regional intra-AD costs, and the second scenario evidences inter-regional costs. We hereafter present the two scenarios.

Scenario 1: Intra-region, intra-availability domain. In this first scenario, we concentrate all the tests in *us-east-1*, an AWS region in USA Ohio. We launch two VMs in the same availability domain (AD) called *us-east-2b*. The two VMs are *t2.micro* with 1v CPU, and 1GB memory. Each VM is launched with default storage capabilities (GP3 with 8 GB storage, 125 Mbps throughput, and 3000 IOPS), and runs Amazon Linux. To evaluate the network cost, we send traffic between the launched VMs using Iperf 3 [14]. This provided an estimate of the data rates required as input for *PricingTheCloud*. In this experiment, we transferred 46.7 GB of data at 111 Mbps for one hour between the client and server VMs, and then reversed the direction of the transfer for a similar duration.

Scenario 2: Inter-region. We launched two VMs, one in east coast USA ((`us-east-2`) called “`VM_us_ohio`”, and the second in the UK (`eu-west-2`) called “`VM_europe_london`”. Both VMs are t2 micro with default characteristics (1v CPU, 1 memory, 8 GB storage). Given the observed asymmetry in network pricing based on traffic origin, we evaluated network costs in both directions: USA to UK and vice versa. This assessment comprised two experimental steps:

Table 1: A summary of the results for estimating compute costs in USD ($). *PricingTheCloud* is the most accurate for the intra-region configuration, while Cloudorado offers better results for the inter-region configuration.

Scenario	Element	AWS Invoices	Pricing TheCloud	Cloudorado	Holori	Vantage
Scenario 1 Intra-Region	VM_ohio_1	0.0100	0.0100	0.0900	0.0110	0.0116
	VM_ohio_2	0.0100	0.0100	0.0900	0.0110	0.0116
	Total ($)	0.0200	0.0200	0.1800	0.0220	0.0232
Scenario 2 Inter-Region	VM_ohio	0.0563	0.0300	0.0900	0.0348	0.0396
	VM_london	0.0600	0.0400	0.0500	0.0396	0.0396
	Total ($)	0.1163	0.0700	0.1400	0.0744	0.0792

Table 2: A summary of the results for estimating network costs in USD. *PricingTheCloud* is the most accurate, while Cloudorado is the only other solution that offers network cost estimation but it is disproportionate.

Scenario	Source	Destination	AWS Invoices	Pricing TheCloud	Cloudo rado	Holori	Vantage
Scenario 1 Intra-AD	`VM1_Ohio`	`VM2_Ohio`	0	0	4.455	–	–
	`VM2_Ohio`	`VM1_Ohio`	0	0	4.455	–	–
		Total ($)	0	0	8.910	–	–
Scenario 3 Inter-Region	`VM_London`	`VM_Ohio`	0.9600	0.9300	8.9100	–	–
	`VM_Ohio`	`VM_London`	1.4900	1.4300	4.4100	–	–
		Total ($)	2.4500	2.3600	13.3200	–	–

1. From UK London to US Ohio: "`VM_europe_london`" is configured as the client, and "`VM_us_ohio`" as the server. We send 45.7 GB of traffic during 1 hours at 109 Mbps bitrate from "`VM_europe_london`" to "`VM_us_ohio`".
2. From US Ohio to UK London: At the end of the previous experiment, "`VM_europe_london`" becomes the server and "`VM_us_ohio`" becomes the client. We now send 71.5 GB of traffic at 85.3 Mbps bitrate. This second experiment is run for 2 hours.

The performance obtained from the above two scenarios is summarized in Tables 1 and 2. Specifically, Table 1 presents compute costs results obtained at the end of each scenario while Table 2 presents the network costs, against the real values obtained from AWS invoices in each case.

We can observe that for scenario 1, the compute costs for each VM involved in the represented architecture are similar. This is because the VMs are in the

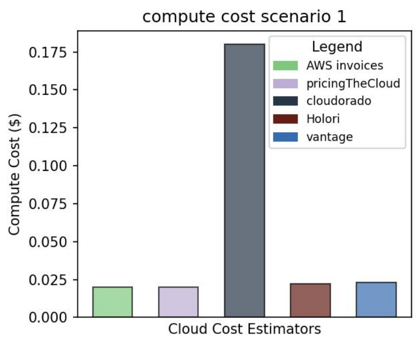

Fig. 1: Compute cost scenario 1

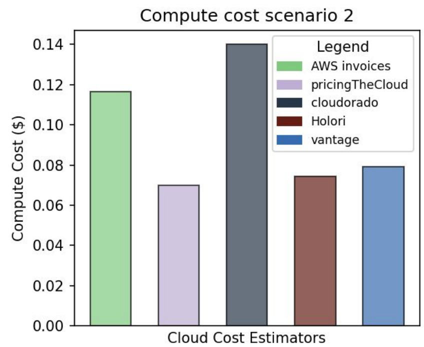

Fig. 2: Compute cost scenario 2

same region and run for the same duration. Moreover, we observe that the total compute costs generated by *pricingTheCloud* are similar to AWS costs because apart from launching the VMs, no other configuration is required. Concerning the network cost, we note that no network costs were generated throughout this experiment by either AWS or *pricingTheCloud* as neither charge for intra-availability domain traffic. In conclusion, our estimator is 100% accurate in predicting compute and network costs for intra-AD configurations.

Results obtained from scenario 2 show that compute costs generated by the AWS invoices are slightly higher ($0.1163) than those generated by *pricingTheCloud* ($0.07). This difference in prices can be explained by the fact that before launching the information exchange process between the two VMs involved in an inter-regional scenario, some configuration need to be applied such as creating a peering connection, configuring IP addresses, setting up a security group, etc. These phases are completed when the VMs are already launched, and thus the compute cost increases slightly due to the time taken for these configurations. In this specific case, *PricingTheCloud* is 60% accurate in predicting the compute costs.Concerning network costs, we observe a difference of $0.09 between AWS-generated network cost value and *pricingTheCloud*-generated value. Our estimator is then 96% accurate in predicting network costs in this case.

The above-presented results show that the developed estimator is 87% accurate on average in predicting compute costs for the two scenarios and 65% accurate on average in predicting network costs. In the following section, we compare the results of our estimator with three state-of-the-art estimators.

5.2 Efficiency Assessment: comparison of *PricingTheCloud*'s generated results with other estimators

Figures 1–4 depict how *PricingTheCloud* performs as compared to three state-of-the-art commercial estimators, namely Cloudorado, Vantage, and Holori. The figures respectively show the total compute costs and the total network costs for each of the aforesaid scenarios.

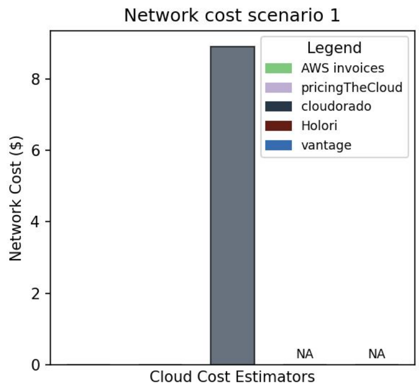

Fig. 3: Network cost scenario 1

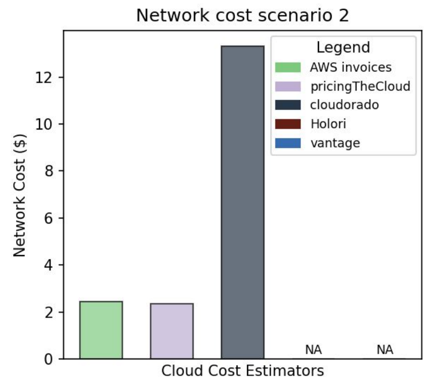

Fig. 4: Network cost scenario 2

As far as total compute costs are concerned, we observe that *PricingTheCloud*, Holori and Vantage generate very similar results for scenario 1, with accuracies of 100%, 90%, and 84%, respectively, when compared to AWS invoices. However, Cloudorado generates a wildly inflated compute cost. For scenario 2 (inter-regional costs), the values generated by Cloudorado, Holori and Vantage are closer to that in AWS invoices.

Specifically, Cloudorado estimates compute cost with an accuracy of 80%, Holori with 64% and Vantage with 68%. However, *PricingTheCloud*'s costs are only 60% accurate and is thus less effective at computing compute costs in the inter-regional scenario.

Turning to network costs, Holori and Vantage are not able to produce any estimates as they do not take this functionality into account. Their missing values are denoted as "–" in the table and "**NA**" in the graphs. Cloudorado's estimates are grossly inflated, similar to its compute costs in scenario 1. *PricingTheCloud* is the most effective at predicting network costs with an accuracy of 100% in scenario 1 and 96% in scenario 2

6 Limitations and Threats to Validity

6.1 Limitations

Our experiments have demonstrated the accuracy and potential of *PricingTheCloud* as a tool for estimating cloud costs. However, like any nascent research, this study has limitations that deserve discussion.

First, we acknowledge that our evaluation focused exclusively on on-demand VMs from AWS, excluding other providers and pricing models. Specifically, we have not yet considered dynamic pricing options such as Spot instances, sustained use discounts, committed resources, or upfront payments. Thus, our current model reflects the specific pricing structures in effect at the time of this study, which may evolve and necessitate updates to the underlying codebase.

Despite these limitations, our work represents a significant step forward in accurate cloud cost estimation, particularly in the crucial area of network costs, which are often neglected by existing tools as shown by our research. Further research will expand the model's scope and enhance its robustness to pricing fluctuations, ultimately providing a more comprehensive and adaptable solution for cloud resource planning.

6.2 Threats to validity

We now discuss potential threats to the validity of our study, particularly in terms of generalizability and trustworthiness.

Construct. The study focused on a specific set of workloads. The estimator's accuracy for workloads with different resource requirements might vary. Future work should investigate a broader range of workloads. Additionally, as discussed under limitations, the study evaluated the estimator on one major CSP. Since cost structures and pricing models can vary significantly across providers, further research is needed to assess generalizability to other cloud platforms.

Internal. The accuracy of the cost estimates depended on invoice data, which serves as the ground truth, ensuring minimal risk of inaccuracy or bias.

External. Similar to any other study, our work has certain limitations, including application architectures. These may not have captured the full spectrum of cloud computing usage scenarios, which could somewhat limit the generalizability of our findings.

7 Conclusion

We have proposed a new estimator to enable companies to estimate their costs before migrating to the cloud. The proposed estimator allows CSPs to have a clear estimate of their costs before migrating to the cloud. Experiments to date have been satisfactory, with an average accuracy of 86.73% for compute costs and 65.44% for network costs in different AWS-to-AWS scenarios. Furthermore, experiments show that the estimator is more efficient than three other available estimators in predicting costs in different scenarios. For future work, we plan to develop the estimation for inter-availability domain network costs and improve storage estimates. We also plan to experiment with other CSPs such as Azure and Google, and explore multi-cloud deployments.

References

1. Cloud Computing Price Comparison | Cloudorado - Find Best Cloud Server from Top Cloud Computing Companies, https://www.cloudorado.com/
2. Holori - End-to-end multi cloud management platform, https://holori.com/
3. Use the BMC Helix Cloud Migration Simulator - BMC Software, https://www.bmc.com/forms/helix-cloud-migration-simulator-trial.html

4. Al Hadwer, A., Tavana, M., Gillis, D., Rezania, D.: A systematic review of organizational factors impacting cloud-based technology adoption using technology-organization-environment framework. Internet of Things **15**, 100407 (2021). https://doi.org/10.1016/j.iot.2021.100407
5. Aldossary, M., Djemame, K.: Energy-based cost model of virtual machines in a cloud environment. In: 5th International Symposium on Innovation in Information and Communication Technology (ISIICT). pp. 1–8 (2018). https://doi.org/10.1109/ISIICT.2018.8613288
6. Brumec, S., Vrček, N.: Cost effectiveness of commercial computing clouds. Information Systems **38**(4), 495–508 (2013). https://doi.org/10.1016/j.is.2012.11.002, special section on BPM 2011 conference
7. Cho, K., Bahn, H.: A cost estimation model for cloud services and applying to pc laboratory platforms. Processes **8**(1) (2020). https://doi.org/10.3390/pr8010076
8. Elhabbash, A., Samreen, F., Hadley, J., Elkhatib, Y.: Cloud brokerage: A systematic survey. Computing Surveys **51**(6), 119:1–119:28 (Jan 2019). https://doi.org/10.1145/3274657
9. Elkhatib, Y., Samreen, F., Blair, G.S.: Same same, but different: A descriptive intra-IaaS differentiation. In: 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid). pp. 690–695 (May 2019). https://doi.org/10.1109/CCGRID.2019.00089
10. Flexera: State-of-the-cloud report 2022. https://info.flexera.com/CM-REPORT-State-of-the-Cloud (Apr 2022)
11. Khan, A.Q., Matskin, M., Prodan, R., Bussler, C., Roman, D., Soylu, A.: Cloud storage cost: a taxonomy and survey. World Wide Web **27**(4), 36 (2024). https://doi.org/10.1007/s11280-024-01273-4
12. Kratzke, N.: Cloud computing costs and benefits. In: Ivanov, I., van Sinderen, M., Shishkov, B. (eds.) Cloud Computing and Services Science. pp. 185–203. Springer New York (2012)
13. Makhlouf, R.R.M.: Cloud Computing: Developing a Cost Estimation Model for Customers. Ph.D. thesis, BTU Cottbus - Senftenberg (Mar 2023). https://doi.org/10.26127/BTUOpen-6284
14. Mortimer, M.: iperf3. https://github.com/thiezn/iperf3-python (2018)
15. Nicolas, R., Pat, M.: EC2 instance timeline, https://instancetyp.es/
16. RENO, N.: cloud infrastructure services market. Tech. rep., SGR (Apr 2022), https://www.srgresearch.com/
17. Samreen, F., Blair, G.S., Elkhatib, Y.: Transferable knowledge for low-cost decision making in cloud environments. Transactions on Cloud Computing **10**, 2190–2203 (Jul 2022). https://doi.org/10.1109/TCC.2020.2989381
18. Tian, J., Elhabbash, A., Elkhatib, Y.: Predicting cloud performance using real-time VM-level metrics. In: International Conference on High Performance Computing and Communications (HPCC). IEEE (Dec 2022). https://doi.org/10.1109/HPCC-DSS-SmartCity-DependSys57074.2022.00184
19. VNTG Inc.: Vantage: Understand your cloud costs. https://www.vantage.sh/