UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

RAFAEL ELTER

Analysing Spot Market Historical Data Across Multiple Regions and Instance Types

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Engineering

Advisor: Prof. Dr. Arthur Francisco Lorenzon

Porto Alegre
February 2024

“If I have seen farther than others,
it is because I stood on the shoulders of giants.”
— Sir Isaac Newton

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my family for their unwavering support throughout my academic journey. I am also immensely thankful to my advisor, Arthur, for his guidance and patience during the final steps of my graduation. Furthermore, I extend my heartfelt appreciation to the Brazilian population for their investment in my education.

ABSTRACT

The spot market of cloud infrastructure services is a relatively recent pricing model made available by cloud providers. This report aims to review the previous works developed regarding this pricing model and perform a historical analysis of the available data. The findings of this work provide insights on the opportunities available by using Spot Instances in multiple regions or with multiple instance types.

Keywords: Spot Market. Cloud Computing. Pricing. Amazon EC2.

Analisando Dados Históricos do Mercado *Spot* para Múltiplas Regiões e Tipos de Instâncias

RESUMO

O mercado *spot* de serviços de infraestrutura de nuvem é um modelo de precificação relativamente recente disponibilizado pelos provedores de nuvem. Este relatório tem como objetivo revisar os trabalhos anteriores desenvolvidos sobre este modelo de precificação e realizar uma análise histórica dos dados disponíveis. Os resultados deste trabalho fornecem insights sobre as oportunidades disponíveis ao usar Instâncias Spot em várias regiões ou com vários tipos de instâncias.

Palavras-chave: spot market, computação em nuvem, precificação, Amazon EC2.

LIST OF ABBREVIATIONS AND ACRONYMS

AZ	Availability Zone
GCP	Google Cloud Platform
AWS	Amazon Web Services
IaaS	Infrastructure as a service
PaaS	Platform as a service
SaaS	Software as a service
SI	Spot Instance
NIST	National Institute of Science and Technology

LIST OF FIGURES

Figure 1.1 Brief history of dynamic pricing mechanisms in the cloud 12

Figure 4.1 AWS *m5.large* Spot Price variation for selected regions 22
Figure 4.2 Azure *A4* Spot Price variation for selected regions 22
Figure 4.3 GCP *c2-standard-4* Spot Price variation for selected regions 23

LIST OF TABLES

Table 4.1 AWS Dataset Sample 21
Table 4.2 Azure Dataset Sample 21
Table 4.3 GCP Dataset Sample 21

Table 5.1 General Purpose | Single Instance Type - Multiple AZs 25
Table 5.2 Compute Optimized | Single Instance Type - Multiple AZs 25
Table 5.3 Memory Optimized | Single Instance Type - Multiple AZs 25
Table 5.4 Storage Optimized | Single Instance Type - Multiple AZs 25
Table 5.5 Group Discount | Single Instance Type - Multiple AZs 25
Table 5.6 General Purpose | Multiple Instance Types - Single AZ 26
Table 5.7 Compute Optimized | Multiple Instance Types - Single AZ 26
Table 5.8 Memory Optimized | Multiple Instance Types - Single AZ 26
Table 5.9 Storage Optimized | Multiple Instance Types - Single AZ 27
Table 5.10 Group Discount | Multiple Instance Types - Single AZ 27
Table 5.11 Group Discount | Multiple Instance Types - Multiple AZs 27

CONTENTS

1 INTRODUCTION....11
2 BACKGROUND....14
2.1 Cloud Computing....14
2.2 Spot Instance Market....15
3 RELATED WORK....18
3.1 Modeling and Prediction of Spot Prices....18
3.2 Bidding Strategy....18
3.3 Spot Market Theoretical Frameworks....19
3.4 Contributions of this work....19
4 METHODOLOGY....20
4.1 Data Acquisition and Preparation....20
4.2 Exploratory Data Analysis....21
4.3 Estimation of opportunities....23
5 RESULTS....24
5.1 Single Instance Type - Multiple AZs....24
5.2 Multiple Instance Types - Single AZ....25
5.3 Multiple Instance Types - Multiple AZs....26
6 CONCLUSION AND FUTURE WORK....28
REFERENCES....29

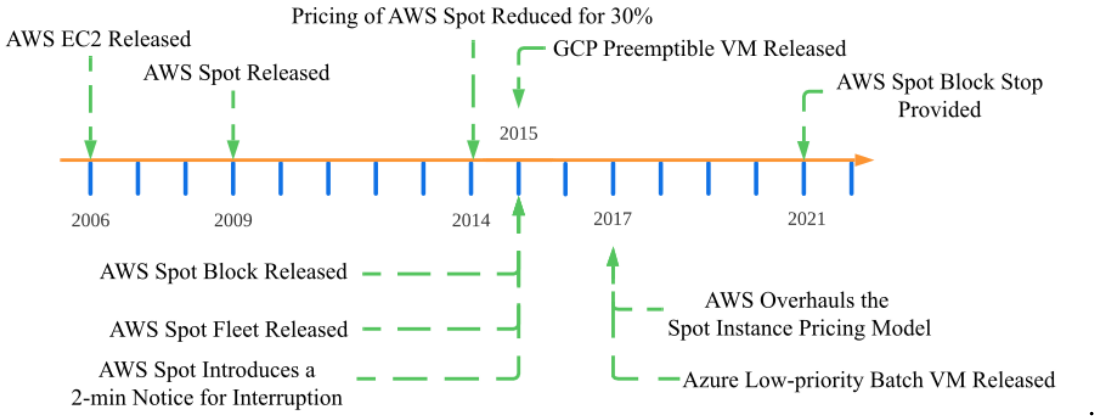
1 INTRODUCTION

In the swiftly evolving realm of cloud computing, the promise of agility, scalability, and cost efficiency has prompted businesses to increasingly migrate their operations to Infrastructure as a Service (IaaS) offerings. Cloud service providers like Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure have facilitated access to on-demand computational resources, resulting in a paradigm shift from traditional IT infrastructure management. However, while the benefits of cloud computing are undeniable, the accompanying operational expenses have emerged as a central concern for organizations striving to optimize their resource allocation while managing costs effectively.

During the nascent stage of cloud computing, when the user and provider counts were modest, static pricing prevailed as the dominant model. However, the landscape has evolved drastically, witnessing a surge in providers and heightened customer expectations, intensifying competition and intricacy. In response to these iterative advancements, cloud computing pioneers were compelled to overhaul their marketing and pricing strategies to optimize profits. The amalgamation of escalating competition and the inadequacies of static pricing, such as challenges in establishing equilibrium pricing, sub-optimal cost-effectiveness, and inflexibility, spurred providers to embrace dynamic pricing. This approach, reliant on the utility and availability of resources, not only cultivates robust competition but also enhances the efficient utilization of resources.

First introduced by AWS in 2009 (AMAZON, 2009) and now available by all major cloud providers, Spot Instances (SIs) are the option for clients to use the non-utilized computing resources of the providers at a lower cost, but without any guarantee of the availability of the machine. A summary of the evolution of SIs can be seen in Figure 1.1, which shows how AWS pioneered this pricing model and how other major cloud providers followed the trend. It is important to note that the behavior of spot prices has changed during this time, mainly to reduce the interruptions of the instances and make the service more appealing to customers used with on-demand pricing schemes. The figure also exemplifies the dynamic environment of the Spot Market in the Cloud, showing recent changes to the model by AWS that provided significant reduction in price volatility and instance availability in 2017 and 2021. These changes provide fresh ground for academics to review work that was previously done and provide newer analysis and models to help decision makers to handle Cloud Expenditure in an effective manner.

Figure 1.1: Brief history of dynamic pricing mechanisms in the cloud



Source: Lin, Pan and Liu (2022)

SIs are offered in an auction system, where clients are bid for the available instances at a discount rate. With their inherently lower costs, these spot instances allow businesses to alleviate the financial burden associated with cloud-based computing. However, their transient nature and variable availability necessitate careful consideration of their utilization. Because of this nature, Amazon categorizes tasks suitable for SIs in four groups (AMAZON, b):

- **Optional tasks**. Not strictly required tasks that can be executed on SIs when prices are low. It can be stopped in case of higher prices.
- **Delayable tasks**. Tasks with deadlines provide flexibility about when they are executed.
- **Acceleratable tasks**. Tasks that can be speeded up by adding more SIs in case of availability of SIs at lower prices.
- **Large scale tasks**. For tasks that may require computing power that one can't access any other way, SIs can cost-effectively run them.

Many companies have shared success stories of migrating tasks from on-demand to SIs and reporting considerable cost savings. Data Processing and Analysis, Scientific Computing, Web Crawling and Scrapping, Background Jobs and Testing, etc., can all be executed with SIs since they possess one or more of the characteristics above. A series of real use cases are described by the providers on their websites, as seen in (AMAZON, a). Furthermore, due to the presented cost reduction and better utilization of resource opportunities, research about SIs is still relevant today, as shown in surveys from (LIN; PAN; LIU, 2022) and (KUMAR et al., 2018).

Therefore, this work aims to explore the data from SIs in search of other oppor-

tunities for price reduction. To achieve this, the goals of this project were to acquire the necessary data from the cloud providers, create theories of optimizations based on an exploratory analysis of the data, and measure the possible gains from those optimizations.

The following sections of this work provide a bibliographical review of related work to this one and a detailed description of how the project was conducted, ensuring the results presented here can be reproducible. It also contains the results obtained during the project, the conclusions, and what can come after this research.

2 BACKGROUND

This section reviews some essential works and concepts developed for utilizing Spot Instances in recent years.

2.1 Cloud Computing

The National Institute of Science and Technology (NIST) defines cloud computing as a model that enables ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction (MELL; GRANCE et al., 2011). This definition highlights several key attributes of cloud computing:

- **On-Demand Self-Service**. Cloud users can provision and manage resources, such as virtual machines, storage, and applications, without requiring direct human intervention from the service provider. This allows for flexibility and agility in resource allocation.
- **Broad Network Access**. Cloud services are accessible over the network, often the internet, from various devices, such as laptops, smartphones, and tablets. This accessibility facilitates remote access and usage.
- **Resource Pooling**. Cloud providers aggregate and pool computing resources to serve multiple users simultaneously. These resources are dynamically allocated and assigned based on demand, achieving efficient utilization.
- **Rapid Elasticity**. Cloud resources can be quickly scaled up or down to accommodate changing workloads. This elasticity ensures that users can access the resources they need when they need them.
- **Measured Service**. Cloud usage is monitored, controlled, and optimized, providing transparency for both providers and consumers. Users are billed based on their usage, often regarding CPU hours, storage, bandwidth, or other relevant metrics.

The NIST further outlines three primary service models within the cloud computing paradigm:

- **Software as a Service (SaaS)**. Consumers use provider-hosted applications acces-

sible over the network. They are relieved from maintenance, patching, and infrastructure management tasks.

- **Platform as a Service (PaaS)**. Consumers deploy applications onto the cloud infrastructure using the provider's programming languages, tools, and services. PaaS offerings facilitate the development and deployment process.
- **Infrastructure as a Service (IaaS)**. Consumers can access virtualized computing resources, such as virtual machines and storage, to deploy and manage their software applications.

Furthermore, NIST identifies four primary deployment models for cloud computing:

- **Public Cloud**. Cloud resources are owned and operated by a third-party provider and made available to the general public over the Internet.
- **Private Cloud**. Cloud resources are exclusively used by a single organization, either managed internally or by a third-party provider.
- **Community Cloud**. Cloud resources are shared by several organizations with common interests, often within a specific industry or domain.
- **Hybrid Cloud**. Cloud resources combine two or more distinct cloud deployment models (public, private, or community), which remain distinct entities but are bound together by technology.

To the time of writing, the business of providing cloud services is mostly concentrated into the three biggest players with it's corresponding market share: AWS (31%), Azure (24%) and Google Cloud Platform (11%). Other players, such as Alibaba Cloud (4%), Salesforce (3%), IBM Cloud (2%) and Tencent Cloud (2%) still compete, but do not have the same strength at the whole sale level (SYNERGY, 2023).

2.2 Spot Instance Market

Cloud providers present pricing models tailored to three distinct instance types to allocate their resources efficiently. For simplicity, the AWS naming of products will be adopted. These models encompass Reserved Instances, On-Demand Instances, and Spot Instances. Users can opt for Reserved Instances when they possess predetermined requirements, securing resources for an entire year, which can be initiated as needed. On

the other hand, On-Demand Instances offer the flexibility to provision resources, catering to fluctuating and uncertain demands dynamically. These instances are allocated hourly, lacking any assurance of availability at any given moment. However, it's worth noting that On-Demand Instances incur higher costs compared to Reserved Instances. Following the conclusion of the reservation period for allocated Reserved Instances, Amazon bills for them akin to On-Demand Instances. Notably, both allocated On-Demand Instances and Reserved Instances persist until the user terminates them.

The third pricing paradigm introduces Spot pricing, under which the instances available are referred to as spot instances. At its core, the spot market functions as an ongoing auction cycle, with the providers determining spot prices. The process involves a perpetual assessment of the residual instances within the Spot pool—a collection of unused Spot Instances that share the same specifications, availability zone, operating system, and network platform. The spot price for these instances aligns with the lowest successful bid in the respective pool. The provider dynamically recalibrates this spot price in response to the fluctuating supply and demand of Spot Instances, with the update frequency ranging from minutes to days.

If sufficient capacity is available, the requested spot instance is initiated only if the user's bid surpasses the hourly spot price that the provider has declared. In simpler terms, if the present spot price exceeds the user's bid, the instance is halted, denoted as an "out-of-bid failure." It's important to highlight that instead of paying the bid price, the user is billed based on the lowest market price stipulated by the provider. Nonetheless, the termination of the instance can be instigated either by the user or the provider. Should the user decide to terminate the instance, all complete hours, including the last partially utilized hour, are included in the calculation. Conversely, the last partial hour isn't factored in if the service provider initiates the instance termination.

In May 2017, Microsoft Azure introduced low-priority VMs, mirroring Amazon EC2's spot instances. Similarly, Google Compute Engine unveiled instances with comparable attributes labeled as preemptible instances. However, unlike Amazon EC2's spot instances, Google's preemptible VMs adhere to fixed pricing, mitigating the need for price projection and minimizing user risk. Additionally, these instances can remain active for up to 24 hours and are subject to a per-second billing approach. (GCP, 2023) (AZURE, 2023)

Due to the potential for eviction at any moment, these instances prove suitable for fault-tolerant and adaptable applications. Consequently, given these attributes, no SLA

guarantees are provided, and service providers can terminate the instance at their discretion. Nonetheless, certain providers do provide a termination warning feature, whereby users receive prior notice of instance reclamation (for instance, a two-minute warning in Amazon EC2) before it occurs.

Given the substantial cost reduction provided by dynamic resource markets within the cloud, creating contemporary frameworks, platforms, and systems that incorporate these instances holds significant importance. Consequently, a substantial amount of research has been devoted to formulating systems and services that leverage these instance types for a variety of applications. (DELDARI; SALEHAN, 2021)

3 RELATED WORK

3.1 Modeling and Prediction of Spot Prices

Because of the auction model adopted for SIs pricing, the price history of VMs under spot provisioning resembles that of a stochastic process or even of a stock listed on a financial exchange. This characteristic has led many researchers to do work trying to model and predict the behavior of SIs prices over time. Knowing the best bid on a specific instance in the future is advantageous since it would allow customers to estimate better how much to bid to keep the resource for the necessary time.

Many have attempted to predict the price of SIs using Machine Learning (ML) methods. Wallace et al. (2013) proposed a Multi-Layer Perceptron (MLP) to predict short-term price variations in the spot market. Arévalos, López-Pires and Barán (2016) used the previous work as a benchmark and to test three other new approaches (Support Vector Poly Kernel Regression, Gaussian Process, and Linear Regression) in a comparative analysis. Fabra, Ezpeleta and Álvarez (2019) created different Linear Regression models based on clusters of Availability Zones that showed similarities between them, besides from providing a good general framework for analyzing SIs. As is regularly the case with complex market data, such as the one from the Spot Market or the common finance market, active research is still persistent to find more efficient ways of predicting it's future movements (KHODAK et al., 2018) (BAUGHMAN et al., 2018) (DOMANAL; REDDY, 2018).

3.2 Bidding Strategy

Other works have focused on determining the optimal bidding strategy for the spot market. (KARUNAKARAN; SUNDARRAJ, 2014) divided bidding strategies into four different categories:

- **S1**. Bidding near to reserved price
- **S2**. Bidding above the average spot price calculated from spot price history
- **S3**. Bidding close to on-demand price
- **S4**. Bidding above the on-demand price

Each approach has its own goals, either to reduce costs or increase the instances'

overall availability. AWS recommends the last strategy to its clients (KUMAR et al., 2018), as they claim it is the best chance to get the desired instance. (KARUNAKARAN; SUNDARRAJ, 2014) makes recommendations for different kinds of applications based on the number of interruptions and mean price from each solution.

3.3 Spot Market Theoretical Frameworks

Another subject for researchers was the proposal of theoretical frameworks for using SIs to reduce operating costs associated with running VMs. (SHARMA et al., 2015) proposed SpotCheck, a cloud platform to interact between clients and IaaS providers, acting transparently to reduce costs without compromising availability. The approach was to use the lower-cost SIs to execute applications and migrate them to on-demand instances whenever the provider revokes the service.

Similarly, (SUBRAMANYA et al., 2015) proposed SpotOn, a batch computing service for the cloud spot market. SpotOn utilizes fault tolerance techniques to exploit spot markets in different service regions and to provide the SLA of an on-demand instance with the discount of SIs (with the performance costs of such a solution).

3.4 Contributions of this work

Due to the constant changes and immaturity of the general spot market, this work aims to present a fresh analysis of the behavior of spot prices utilizing current datasets. It also aims to provide top-level guidance for future works, speeding up the understanding of the dynamics of this market relating to its core variables: instance type, region, and time.

4 METHODOLOGY

This project aims to find and measure ways to reduce costs with spot instances. For this, it was first necessary to obtain the available data from SIs providers (spot price, on-demand price, etc.). Then, an exploratory analysis was conducted to better understand the behavior of the obtained time series and to elaborate theories on how optimizations to use SIs could be done. For last, the possible gains of these optimizations were estimated as a way to measure how significant the opportunities are.

4.1 Data Acquisition and Preparation

The data used in this work comes from the work of Kim et al. (2023), which contained historical data for Spot Instance pricing from the three major IaaS providers (AWS, Azure, and GCP). This previous work consisted of collecting data from multiple providers and providing it to other researchers to speed up cloud system research to improve spot instance usage and availability while reducing cost. The used dataset contained a full year of data, starting from October 2022. To work with the provided data, a step of preparation was required. The main problems were the file format in which data was originally stored and the changes made to the data structure during the period.

The data came in Comma Separated Values (CSV), which, although a suitable format for human readability, cannot be considered space or speed-efficient. For those reasons, the data was transformed to the Apache Parquet data storage format, which, besides being efficient, is free and open-sourced. The exploratory process was much quicker with the data stored as Parquet files. It allowed the project to be executed on local machines (rather than in special purposes ones).

Since the original data comes from distinct sources, each provider dataset contains its columns (i.e., they share some but not all). All providers' data points contain the timestamp, instance type, region, spot price, and on-demand price. Tables 4.1, 4.2, and 4.3 provide samples of the available data from the major cloud providers AWS, Azure, and GCP, respectively. As seen in table 4.1, AWS's pricing is more granular than it's competitors since it provides different prices for each Availability Zone (AZ). For this reason, when referring to AWS's data the presented results are using the AZ of the instance, while with the other providers the region is used.

Table 4.1: AWS Dataset Sample

Time	AZ	Instance	SPS	IF	Price	Spot Price
2022-10-01 00:00:00	use1-az1	m5.large	3	2	0.096	0.038
2022-10-01 00:10:00	use1-az1	m5.large	3	2	0.096	0.038
2022-10-01 00:20:00	use1-az1	m5.large	3	2	0.096	0.038
2022-10-01 00:30:00	use1-az1	m5.large	3	2	0.096	0.038
2022-10-01 00:40:00	use1-az1	m5.large	3	2	0.096	0.038

Source: Author

Table 4.2: Azure Dataset Sample

Time	Region	Tier	Instance	Price	Spot Price
2022-10-01 00:00:00	US East 2	Standard	A4	0.480	0.070
2022-10-01 00:00:00	US East 2	Basic	A4	0.352	0.052
2022-10-01 00:10:00	US East 2	Standard	A4	0.480	0.070
2022-10-01 00:10:00	US East 2	Basic	A4	0.352	0.052
2022-10-01 00:20:00	US East 2	Standard	A4	0.480	0.070

Source: Author

Table 4.3: GCP Dataset Sample

Time	Region	Instance	Price	Spot Price
2022-10-01 00:00:00	us-east1	c2-standard-4	0.209	0.021
2022-10-01 01:00:00	us-east1	c2-standard-4	0.209	0.021
2022-10-01 02:00:00	us-east1	c2-standard-4	0.209	0.021
2022-10-01 03:00:00	us-east1	c2-standard-4	0.209	0.021
2022-10-01 04:00:00	us-east1	c2-standard-4	0.209	0.021

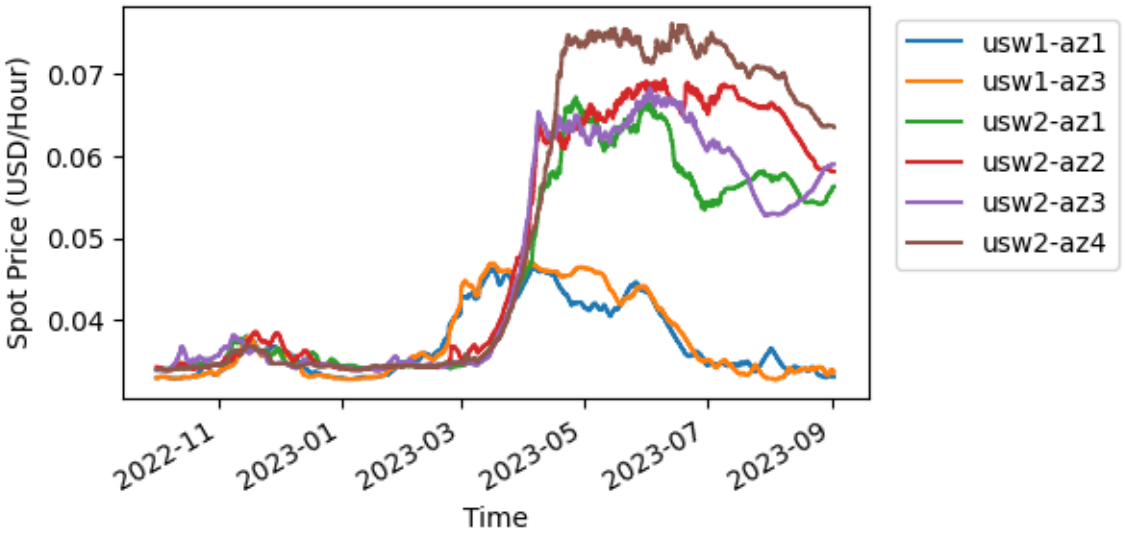
Source: Author

4.2 Exploratory Data Analysis

In order to assess what opportunities are available with the spot market oscillations, first, it was necessary to determine how these markets behave. For this, the spot price time series of selected instances were plotted to map the behavior. Figures 4.1, 4.2 and 4.3 provide the variation of Spot Price for a single instance across selected Availability Zones/Regions, each one represented by a different color. Each time series contains a frequency of 1 data point for each 10 minutes. From observing the plots, the following conclusions can be made:

- AWS's prices are the only ones that resemble a market, where supply and demand define the price.
- The prices from Azure and GCP move in jumps, which was expected from GCP, since it states that its prices are adjusted only monthly, but not from Azure, making no such claims in its documentation.

Figure 4.1: AWS *m5.large* Spot Price variation for selected regions

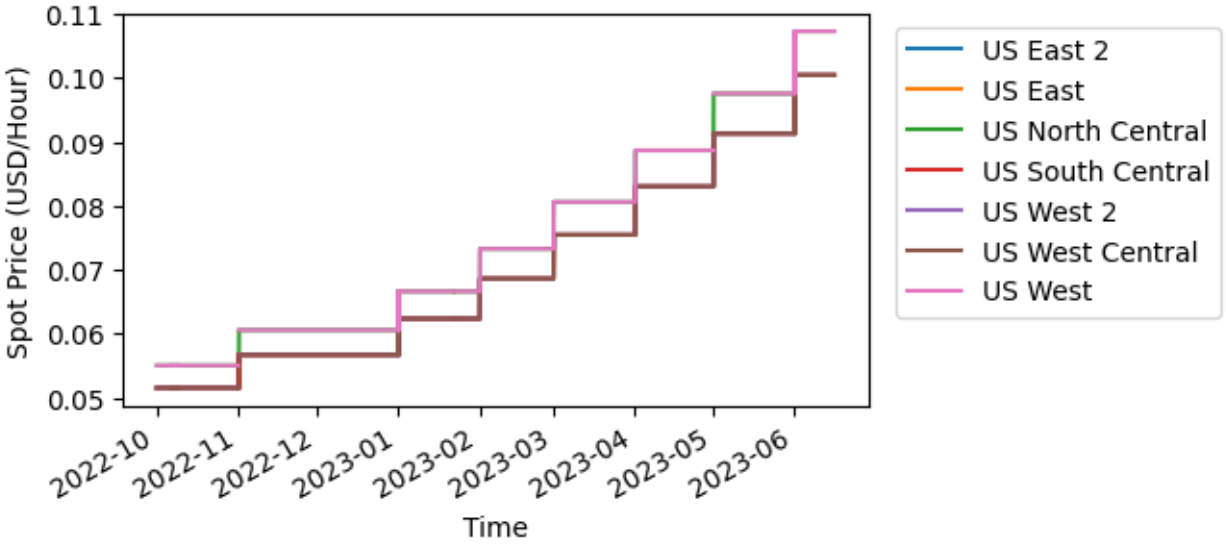


Source: Author

- The difference in price between regions in Azure is static. The mean Pearson Correlation between the same instance types of different regions is approximately 1.

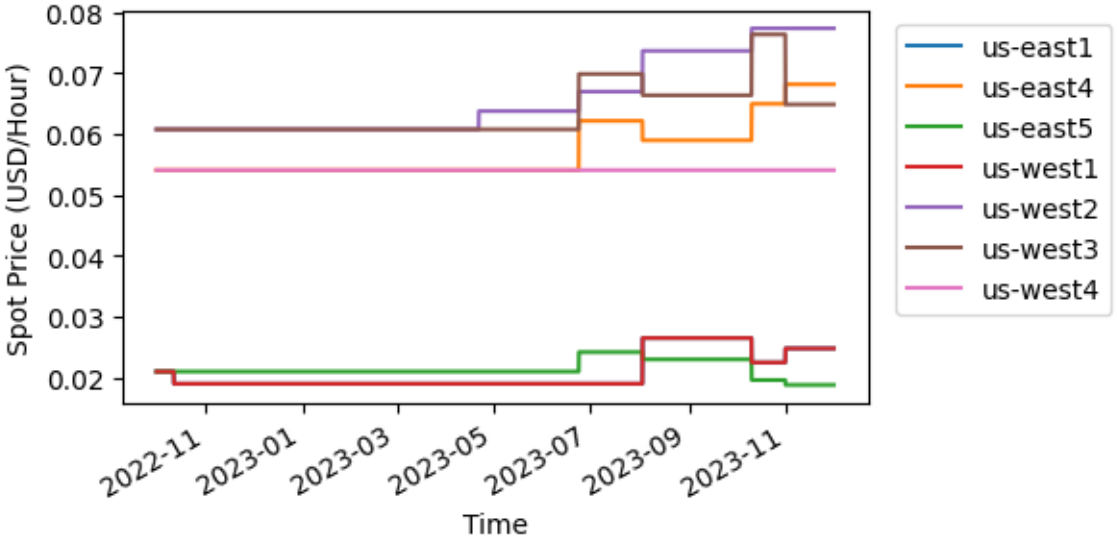
During this phase of the work, it was noticeable how the prices from AWS showed greater variability than the rest and, thus, showed more significant promise of the opportunities that motivated the analysis. For this reason, the work followed on examining only AWS's data.

Figure 4.2: Azure *A4* Spot Price variation for selected regions



Source: Author

Figure 4.3: GCP *c2-standard-4* Spot Price variation for selected regions



Source: Author

4.3 Estimation of opportunities

From the observed results of the exploratory analysis phase, it was possible to conclude that savings could be achieved by allowing the application to be run in any available region. To measure how significant these savings could be, an estimation was made by comparing the mean price of an instance for multiple regions, which intended to represent the choice of choosing a region/AZ and staying with it, against the lowest price available for each data point, representing what ideally could be achieved by changing the application's region with the price fluctuations.

As a continuation, it was also tested to verify if savings could be achieved by allowing the application to be run in different instance types (in addition to different regions). However, this analysis can be much more complicated than the previous one since different instances imply significantly different execution times. For simplification, this factor was not considered during this work, and the analysis was used as the basis for the cheapest option available between a group of instances, as measured by the on-demand price. Doing so allows us to work with an approximated worst-case scenario and make reasonable claims.

5 RESULTS

Since AWS offers its clients a significant variety of instance types, only some of those instances were selected for further analysis. This choice was made based on AWS's classification, which splits the instance types into five groups: General Purpose, Compute Optimized, Memory Optimized, Accelerated Computing, and Storage Optimized. Three instance types were selected for each category based on AWS's catalog of Intel Processor-based instances. This last criterion was selected to enable future research beyond the work done here, which will be discussed in greater depth in the last chapter.

The analysis was also made using only regions and AZs inside the United States. It was done so in order to compare comparable prices and to maintain networking latency negligible. The United States was selected since it contains the cheapest cloud prices in the world and hosts most of the worlds cloud applications.

The following sections contain the results derived from the data analysis. In the tables presented in this chapter, all the prices are stated in US Dollars per hour, the standard for IaaS, and the discounts are stated in percentages. The results are separated into three distinct sections, each one representing a different scenario. In all scenarios, it's assumed that the customer can freely change his allocation at any given time.

5.1 Single Instance Type - Multiple AZs

The first scenario simulates the possible discounts from being free to pick any given AZ to run the desired application, given that the instance type selected remains the same. Relating this scenario to the real world relates to a customer with an application running on an SI, with a specific instance type, and in a specific AZ. This customer, who does not have network latency constraints, now has the opportunity to pick any AZ to run his application, being free to choose the cheapest option at all times.

Tables 5.1, 5.2, 5.3 and 5.4 display what the calculated mean price for an instance using multiple AZs and the calculated best price that could be achieved at all times. The tables also indicate the discount between those two prices, showing the magnitude of the difference between them. Table 5.5 shows the aggregated results from all groups.

Table 5.1: General Purpose | Single Instance Type - Multiple AZs

Instance Type	Mean Price	Best Price	Discount
t2.large	0.0434	0.0303	30.15
t3.large	0.0397	0.0263	33.77
m5.large	0.0441	0.0268	39.21

Source: Author

Table 5.2: Compute Optimized | Single Instance Type - Multiple AZs

Instance Type	Mean Price	Best Price	Discount
c5.large	0.0431	0.0283	34.23
c5n.large	0.0459	0.0254	44.63
c6i.large	0.045	0.0264	41.27

Source: Author

Table 5.3: Memory Optimized | Single Instance Type - Multiple AZs

Instance Type	Mean Price	Best Price	Discount
r5.large	0.0428	0.0287	32.99
r6i.large	0.0458	0.0282	38.52
z1d.large	0.0733	0.0591	19.37

Source: Author

Table 5.4: Storage Optimized | Single Instance Type - Multiple AZs

Instance Type	Mean Price	Best Price	Discount
i3.large	0.0543	0.0458	15.66
i4i.large	0.0587	0.0507	13.55
i3en.large	0.0695	0.0655	5.74

Source: Author

Table 5.5: Group Discount | Single Instance Type - Multiple AZs

Instance Group	Average Discount
General Purpose	34.38
Compute Optimized	40.04
Memory Optimized	30.3
Storage Optimized	11.65

Source: Author

5.2 Multiple Instance Types - Single AZ

The second scenario illustrates the opposite case from the previous one. Now, a customer with an application running on an SI with a specific instance type and in a specific AZ can run the same application in any of the selected instance types. This analysis imposes a challenge that was not seen in the previous section. Running an application in different instance types incurs in different execution times, impacting the final bill since

the prices are given by the hour. To incorporate this effect, tables 5.6, 5.7, 5.8 and 5.9 also contain a column for the mean price of keeping the cheapest option at all times. Table 5.10 shows the aggregated results from all groups.

Table 5.6: General Purpose | Multiple Instance Types - Single AZ

AZ	Price Cheapest	Price Average	Price Best	Discount to Average	Discount to Cheapest
usw1	0.0340	0.0350	0.0319	9.00	6.36
usw2	0.0447	0.0455	0.0382	15.98	14.41
use1	0.0510	0.0499	0.0406	18.71	20.48
use2	0.0330	0.0300	0.0242	19.39	26.73

Source: Author

Table 5.7: Compute Optimized | Multiple Instance Types - Single AZ

AZ	Price Cheapest	Price Average	Price Best	Discount to Average	Discount to Cheapest
usw1	0.0358	0.0333	0.0311	6.51	13.05
usw2	0.0469	0.0507	0.0436	13.98	6.96
use1	0.0503	0.0528	0.0406	23.12	19.29
use2	0.0308	0.0305	0.0288	5.6	6.56

Source: Author

Table 5.8: Memory Optimized | Multiple Instance Types - Single AZ

AZ	Price Cheapest	Price Average	Price Best	Discount to Average	Discount to Cheapest
usw1	0.0354	0.0449	0.0343	23.59	3.05
usw2	0.0492	0.0600	0.0461	23.21	6.34
use1	0.0473	0.0573	0.0364	36.42	22.93
use2	0.0316	0.0414	0.0284	31.22	9.98

Source: Author

5.3 Multiple Instance Types - Multiple AZs

This last section merges both previous ones and considers that all instance types from each group could be selected in any given instant. The found results are displayed in table 5.11. Crossing these results with the ones from the previous sections, it can be seen that most of the available discount comes from the lack of an AZ constraint. It also demonstrates that the benefits of removing each constraint are not fully correlated, since the final result is better than the previous ones.

Table 5.9: Storage Optimized | Multiple Instance Types - Single AZ

AZ	Price Cheapest	Price Average	Price Best	Discount to Average	Discount to Cheapest
usw1	0.0516	0.0611	0.0516	15.55	0.00
usw2	0.0563	0.0617	0.0547	11.34	2.88
use1	0.0579	0.0630	0.0556	11.75	3.95
use2	0.0465	0.0551	0.0461	16.32	0.92

Source: Author

Table 5.10: Group Discount | Multiple Instance Types - Single AZ

Instance Group	Average Discount to Mean	Average Discount to Cheapest
General Purpose	15.77	17.00
Compute Optimized	12.30	11.46
Memory Optimized	28.61	10.58
Storage Optimized	13.74	1.94

Source: Author

Table 5.11: Group Discount | Multiple Instance Types - Multiple AZs

Group	Price Cheapest	Price Average	Price Best	Discount to Average	Discount to Cheapest
General Purpose	0.0424	0.0242	0.0434	42.93	44.26
Compute Optimized	0.0447	0.0252	0.0431	43.55	41.5
Memory Optimized	0.053	0.0275	0.0428	48.15	35.74
Storage Optimized	0.0607	0.0458	0.0543	24.53	15.71

Source: Author

6 CONCLUSION AND FUTURE WORK

From the results seen earlier in this work, the conclusion is that significant gains can be extracted by adding the possibility of utilizing multiple AZs and multiple instance types, in ranges from 15% to 45% according to the estimations. While an approach exploiting this fact certainly cannot apply to many use cases, such as where the stability of the service is a must, there are scenarios where an approach to take advantage of the found results could be feasible. As presented earlier, these scenarios include but are not limited to, CI/CD pipelines and software testing. To take advantage of the findings, one must allocate SIs and constantly reevaluate the current price.

Further works could improve the ideas presented here in many ways. In case the scenario changes, a multi-provider analysis could be of great value. The implementation part of everything presented here could cement the findings and provide empirical evidence of cost reduction opportunities. The latter is a must since there is a huge gap between theory and practice when complex systems are involved. This work did not consider engineering costs to build a solution, neither the data transfer rates charged by providers nor the difference in processing speed for different instance types as presented by O'Loughlin and Gillam (2014). These items can change the magnitude of the findings from this work. To minimize this chance, findings from empirical studies analyzing the investment in Cloud Infrastructure, such as Kotas, Naughton and Imam (2018), can be brought to the Spot Market, which, with its constant changes and improvements, is still fertile ground for research.

REFERENCES

AMAZON. **Amazon EC2 Spot - Customers - Amazon Web Services — aws.amazon.com**. <https://aws.amazon.com/pt/ec2/spot/customers>. [Accessed 04-02-2024].

AMAZON. **AWS Product and Service Pricing | Amazon Web Services — aws.amazon.com**. <https://aws.amazon.com/pricing>. [Accessed 18-08-2023].

AMAZON. **Announcing Amazon EC2 Spot Instances**. 2009. <https://aws.amazon.com/about-aws/whats-new/2009/12/14/announcing-amazon-ec2-spot-instances/>. [Accessed - 28-01-2024].

ARéVALOS, S.; LóPEZ-PIRES, F.; BARáN, B. A comparative evaluation of algorithms for auction-based cloud pricing prediction. In: **2016 IEEE International Conference on Cloud Engineering (IC2E)**. [S.l.: s.n.], 2016. p. 99–108.

AZURE. **Spot Virtual Machines – Spot Pricing and Features | Microsoft Azure — azure.microsoft.com**. 2023. <https://azure.microsoft.com/en-us/products/virtual-machines/spot/>. [Accessed 18-08-2023].

BAUGHMAN, M. et al. Predicting amazon spot prices with lstm networks. In: **Proceedings of the 9th Workshop on Scientific Cloud Computing**. New York, NY, USA: Association for Computing Machinery, 2018. (ScienceCloud'18). ISBN 9781450358637. Available from Internet: <https://doi.org/10.1145/3217880.3217881>.

DELDARI, A.; SALEHAN, A. A survey on preemptible iaas cloud instances: challenges, issues, opportunities, and advantages. **Iran Journal of Computer Science**, Springer, v. 4, p. 1–24, 2021.

DOMANAL, S. G.; REDDY, G. R. M. An efficient cost optimized scheduling for spot instances in heterogeneous cloud environment. **Future Generation Computer Systems**, v. 84, p. 11–21, 2018. ISSN 0167-739X. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0167739X17303667>.

FABRA, J.; EZPELETA, J.; ÁLVAREZ, P. Reducing the price of resource provisioning using ec2 spot instances with prediction models. **Future Generation Computer Systems**, v. 96, p. 348–367, 2019. ISSN 0167-739X. Available from Internet: <https://www.sciencedirect.com/science/article/pii/S0167739X1831166X>.

GCP. **Preemptible VM instances | Compute Engine Documentation | Google Cloud — cloud.google.com**. 2023. <https://cloud.google.com/compute/docs/instances/preemptible>. [Accessed 18-08-2023].

KARUNAKARAN, S.; SUNDARRAJ, R. P. Bidding strategies for spot instances in cloud computing markets. **IEEE Internet Computing**, IEEE, v. 19, n. 3, p. 32–40, 2014.

KHODAK, M. et al. Learning cloud dynamics to optimize spot instance bidding strategies. In: **IEEE INFOCOM 2018 - IEEE Conference on Computer Communications**. [S.l.: s.n.], 2018. p. 2762–2770.

KIM, K. et al. Public spot instance dataset archive service. In: **Companion Proceedings of the ACM Web Conference 2023**. New York, NY, USA: Association for Computing Machinery, 2023. (WWW '23 Companion), p. 69–72. ISBN 9781450394192. Available from Internet: <https://doi.org/10.1145/3543873.3587314>.

KOTAS, C.; NAUGHTON, T.; IMAM, N. A comparison of amazon web services and microsoft azure cloud platforms for high performance computing. In: IEEE. **2018 IEEE International Conference on Consumer Electronics (ICCE)**. [S.l.], 2018. p. 1–4.

KUMAR, D. et al. A survey on spot pricing in cloud computing. **Journal of Network and Systems Management**, Springer, v. 26, p. 809–856, 2018.

LIN, L.; PAN, L.; LIU, S. Methods for improving the availability of spot instances: A survey. **Computers in Industry**, Elsevier, v. 141, p. 103718, 2022.

MELL, P.; GRANCE, T. et al. The nist definition of cloud computing. Computer Security Division, Information Technology Laboratory, National . . . , 2011.

O'LOUGHLIN, J.; GILLAM, L. Performance evaluation for cost-efficient public infrastructure cloud use. In: ALTMANN, J.; VANMECHELEN, K.; RANA, O. F. (Ed.). **Economics of Grids, Clouds, Systems, and Services**. Cham: Springer International Publishing, 2014. p. 133–145. ISBN 978-3-319-14609-6.

SHARMA, P. et al. Spotcheck: Designing a derivative iaas cloud on the spot market. In: **Proceedings of the Tenth European Conference on Computer Systems**. [S.l.: s.n.], 2015. p. 1–15.

SUBRAMANYA, S. et al. Spoton: a batch computing service for the spot market. In: **Proceedings of the sixth ACM symposium on cloud computing**. [S.l.: s.n.], 2015. p. 329–341.

SYNERGY, R. G. **AI Helps to Stabilize Quarterly Cloud Market Growth Rate; Microsoft Market Share Nudges Up Again | Synergy Research Group — srgresearch.com**. 2023. <https://www.srgresearch.com/articles/ai-helps-to-stabilize-quarterly-cloud-market-growth-rate-microsoft-market-share-nudges-up-again>. [Accessed 06-02-2024].

WALLACE, R. M. et al. Applications of neural-based spot market prediction for cloud computing. In: **2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)**. [S.l.: s.n.], 2013. v. 02, p. 710–716.