Strategic Management of Multi Cloud Adoption: A Framework for Decision-making and Governance

MSc Research Project

Cloud Computing

Sachin Reddy Chidurala

Student ID: X22140476

School of Computing

National College of Ireland

Supervisor: Dr. Punit Gupta

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Sachin Reddy Chidurala		
Student ID:	x22140476		
Programme:	Cloud Computing	**Year:**	2023
Module:	MSc Research Project		
Supervisor:	Dr. Punit Gupta		
Submission Due Date:	14-12-2023		
Project Title:	Strategic Management of Multi Cloud Adoption: A Framework for Decision-making and Governance		
Word Count:	6452	**Page Count**:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Sachin Reddy Chidurala

Date: 14-12-2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on the computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Strategic Management of Multi Cloud Adoption: A Framework for Decision-making and Governance

Sachin Reddy Chidurala

x22140476

Abstract

This report discusses and presents the study on optimization algorithms for multi-cloud task allocation (bat algorithm and original ant lion optimizer). Systematic CloudSim-based simulation, with quantifiable considerations of performance and cost-effectiveness metrics, have been considered for the evaluation of proposed dynamic task distribution among virtual machines. BAT Algorithm gave better overall execution cost than the ALO algorithm. The study examines the nature of such a solution as opposed to any other cost-minimising or computational-efficient compromise that perhaps, a decision-maker might find most appropriate. These results will directly benefit efficient resource utilisation and lower costs for actual multi-cloud implementations. This report also makes recommendations on improving further algorithms, adding additional hybrid techniques, features, and self-adaptive loading of loads. This work can serve as a foundational step for better optimization of cloud resources on heterogeneous computing platforms.

1 Introduction

1.1 The Rise of Multi-Cloud Adoption

In this era of digitization, marked by swift innovation and never-ending changes, organizations are progressively leveraging the fluidity and expandability of cloud computing to meet their dynamic requirements. Thus, there has been a shift towards the multi-cloud adoption paradigm. Today, businesses use various cloud platforms to enhance speed, affordability, and reliability.

Switching to multi-cloud is crucial and here are a few major ones. One of the issues arising due to relying on a single vendor is vendor lock and it can be eliminated by this approach (Raj and Surianarayanan, 2021b). Moving the infrastructure of the organisation to different cloud platforms allows more agility and flexibility in the management of the IT resources. The organisation's freedom allows it to choose the best cloud provider according to a specific workload and to avoid the offerings of a particular vendor (Chatzithanasis *et al.*, 2021b).

An organisation also can use multi-cloud to leverage different service providers' unique strengths and advantages. Cloud providers have various aspects of strengths and weaknesses depending on high-performance computing, efficient storage or intelligent analysis. Companies can use different cloud platforms and pick the best service for every workload that enhances their performance and productivity (Hong *et al.*, 2019b).

Finally, it enhances resiliency and disaster recovery in multi-cloud setups. With multiple cloud providers hosting on different locations, the risk of an outage or service interruption of one provider is mitigated. This ensures increased reliability and business reliability to minimise any failure possibility time and output loss due to failure. Also, organisations can easily recover their critical operations with multi-cloud deployment because the failure is also localised (Hong *et al.*, 2019b).

1.2 Challenges of a Multi-Cloud Adoption

While the benefits of multi-cloud adoption are significant, managing a complex environment with multiple cloud providers presents its own set of challenges. These challenges include:

- **Cost management:** It is rather challenging to optimise costs across multiple cloud providers due to the intricacy of tracking, monitoring, and optimization tools. Organisations must clearly understand their cloud use patterns and expenses per platform to detect the ways money can be saved and optimise existing spending strategies (Chatzithanasis *et al.*, 2021b).
- **Workload allocation:** The placement of workloads in the best way possible in different cloud platforms is very important in influencing performance and cost. Such an analysis is very demanding and involves many aspects such as resource requirements, data dependency and network connectivity among different cloud platforms. Good workload allocation helps put each work into the most appropriate cloud, thereby ensuring maximum efficiency, and minimising unnecessary expenditure (Hong *et al.*, 2019b).
- **Decision-making complexity:** Appropriate cloud platforms for each workload should consider performance, cost, security, compliance, and service offering aspects. In this regard, making an instantaneous decision may require scrutinising each cloud provider's capability against the workload specifications (Yeganeh *et al.*, 2020).
- **Governance and control:** Ensuring the advantages of cloud agility and centralised governance are harmonised is difficult, and even more so if several cloud platforms are used. This means setting out guidelines on how to manage access, security, compliance, and the use of resources across multiple cloud infrastructures. Proper management entails uniformity in organisational policy adherence, management of risk, and conformance to statutory obligations (Raj and Surianarayanan, 2021b).

1.3 Addressing the Challenges: A Framework for Strategic Management

It calls for establishing a mechanism for strategic management for the true potential of multi-cloud adoption to be realised in an organisation. A complete framework with a cloud-based workload allocation approach for making decisions and cost-effective plans must be developed.

The study aims to contribute to this critical area, and it will present a new way of supporting multi-cloud usage. The multiple clouds management model is based on cloud simulation data, cost analysis, and nature-based optimization algorithms for the development of a comprehensive and realistic model. This research addresses the challenges of multi-cloud adoption by providing:

- **A structured approach for cloud selection and workload allocation:** The performance standards, cost models, security requirements and compliance aspects that will form the basis of an optimised workload distribution among different cloud environments (Wang, Z., Hayat).
- **Advanced cost-management tools and strategies:** By using this, organisations can track, analyse, and measure the cloud usage patterns across different platforms, and look for options to cut costs but enhance spending on the cloud resources.
- **Data-driven insights through cloud simulation:** This is the process where one uses these simulation tools to simulate different scenarios to optimise them.

1.4 Research Objective

The primary objective of this research is to design and implement optimization algorithms for reducing execution cost and runtime in a multi-cloud model. And assess the performance of the proposed framework using computer-based simulations. Developing a new multi-cloud adoption strategy management based on cost analysis, and optimization algorithms.

2 Related Work

The research on multi-cloud adoption has rapidly expanded alongside its growing adoption in industry. Here this section provides a comprehensive review of relevant literature focusing on:

2.1 Decision-making Models and Frameworks for Cloud Selection and Workload Allocation

A Self-Optimized Generic Workload Prediction Framework for Cloud Computing by (Vinodh Kumaran Jayakumar). The research reviews the challenges of load prediction in cloud environments and introduces LoadDynamics, a brand-new general approach. To achieve high precision, LoadDynamics uses LSTM models and internally optimizes such parameters for each workload. Various workload traces were used for evaluating the framework and it showed much smaller prediction errors in comparison with current solutions.

Using LoadDynamics-enabled auto-scaling with Google Cloud resulted in shorter times for jobs to finish as well as better utilisation of virtual machines compared to most other predictors that exist for this purpose. The paper describes the design of LoadDynamics, its thorough evaluation and a case study on its use by the public cloud. LoadDynamics provides an innovative approach to precise workload prediction and autoscaling in the cloud environment.

2.2 Cost Optimization Strategies for Multi-Cloud Environments

Exploring cost-efficient bundling in a multi-cloud environment by Chatzithanasis Georgios, Filiopoulou Evangelia, Michalakelis Christos and Nikolaidou Maria. The research talks about the advantages and disadvantages of hybrid-cloud and multi-cloud environments for business. It also focuses on the benefits of utilising hybrid cloud which entails risk management, operational efficiency, and avoidance of locking in into a single vendor.

It also emphasises the benefits and cost savings of using multi-cloud (Jiang, F., Ferriter, K. and Castillo). This article suggests a technique for analysing multi-cloud efficiency through DEA. Twenty-three IaaS providers' pricing strategies are analysed using the DEA methodology. These findings affirm the assertion that it is possible to derive economies of scale and cost savings using the multi-mode model, and the paper concludes that the MCC approach greatly improves overall cloud performance. Finally, it points out possible fields for further investigations on issues of non-functional parameters, or techno-economic analysis from the CB services' perspective.

2.3 Governance Models for Managing Multi-Cloud Deployments

Model-based deployment of secure multi-cloud applications by (Valentina Casola). The research outlines the problems associated with multi-cloud apps and proposes the MUSA framework as an SBD solution for applying such apps. SSLA-driven security DevOps for cloud applications development. MUSA deployers describe the models and tools that separate the multi-cloud application modelling from applications deployment and provision of cloud services. The deployer must prepare a deployment plan and automatic configuration of all the components that support multi-cloud applications across CSPs.

Also, the paper offers an innovative model-oriented technique of provisioning and organizing software elements in a multi-cloud scenery accompanied by a case study regarding genuine planning of air travel apposition. In summary, this paper highlights a complete approach for applying the implementation stage of multi-cloud apps.

2.4 Cloud Simulation Tools and their Application in Multi-Cloud Research

The research paper "Multi-Cloud: A Comprehensive Review by Hamza Ali Imran" gives an overview of the development stages and outcomes of clouds, paying attention to multi-cloud environments. The paper discusses issues related to cross-over among consumers as they move around in cloud service. In this research, scalability and affordability advantages provided by cloud-based software for use in different contexts are highlighted.

Specifically, those include educational and instructional facilities and market needs. In addition, it assesses the functionalities, risks, and advantages of using multi-cloud systems that are determined by experiments. Lastly, the paper provides models and mechanisms for selecting, administering, and protecting multi-cloud platforms and their application in bioinformatics and simulations. Several ways of achieving a multi-cloud platform are proposed and their effects on performance, security, and high-performance computing.

2.5 Gaps in Existing Research and Potential Future Directions

The existing research on multi-cloud adoption provides valuable insights and frameworks, but still, some gaps need to be addressed:

Integration of cloud simulation and optimization algorithms: Most existing research works use cloud simulation tools and optimization algorithms individually. However, there is an important need to develop new integrated frameworks that consist of these methodologies to deliver a more comprehensive and based-on-the-facts approach to multi-cloud administration.

Consideration of security and compliance requirements: Most prevailing models emphasize efficiency and expenses, overlooking vital factors of safety and conformity. Further research must, therefore, build frameworks that incorporate their consideration in the making of decisions as well distribution of tasks.

3 Research Methodology

Cloud computing is a way through which one can access servers, data storage, databases, networks, software, analytics, and intelligence as computing resources and services. Some advantages provided by cloud computing include scalability, adaptability, dependability, security, and affordability. However, cloud pricing is very complex and depends on factors like service type, location, data usage, and discounts (Al-Roomi, Al-Ebrahim). The purpose of this section is to examine and contrast the price structures of three prominent cloud service providers: GCP, AWS, and Microsoft Azure. This section explains how data collected in November 2023 captures the basic cost of computing, storage and network services in Europe (Ireland) and under the Linux operating system.

3.1 Computation Services

Compute services comprise essential fundamental services that give the required processing powers and memory to run the applications/workloads in the cloud. The three cloud service providers have their specific VM instances that range in computing power, memory capacity, disk space, and bandwidth. Price is dependent on the operating system adopted, the mode of payment, and on the length of the contract in place. Table 3.1 shows the price for instance in Ireland (EU including the UK) in Nov 2023.

Instance type	AWS	Azure	Google Cloud
General purpose	€0.072 for t3.xlarge (4 vCPU, 16 GB RAM)	€0.144 for D4s v3 (4 vCPU, 16 GB RAM)	€0.083 for n1-standard-4 (4 vCPU, 15 GB RAM)
Compute optimized	€0.256 for c5.xlarge (4 vCPU, 8 GB RAM)	€0.156 for F4s v2 (4 vCPU, 8 GB RAM)	€0.104 for c2-standard-4 (4 vCPU, 16 GB RAM)
Memory optimized	€0.144 for r5.xlarge (4 vCPU, 32 GB RAM)	€0.168 for E4s v3 (4 vCPU, 32 GB RAM)	€0.138 for n1-highmem-4 (4 vCPU, 26 GB RAM)
GPU	€0.68 for g4dn.xlarge (4 vCPU, 16 GB RAM, 1 GPU)	€0.555 for NV4as v4 (4 vCPU, 14 GB RAM, 1 GPU)	€0.539 for n1-standard-4 + 1 NVIDIA Tesla T4 GPU

Table 3.1 Hourly prices for compute services in the EU (Ireland) region and Linux operating system

Table 3.1 reveals great variance in the three-cloud provider's compute service pricing. However, AWS and Google Cloud usually cost less for all instance types except for the GPU instances where Azure beats them. Other factors that affect pricing include operating system, billing type, and commitment length. AWS and Azure offer savings on reserved instances that are bought for the fixed durations of e.g., one year or even three years.

To help businesses get discounts, Google Cloud only charges committed usage instances that involve the purchase of a certain quantity of virtual CPU and memory within a one to three-year period. Google Cloud also offers discounts for instances used for a large part of the month known as continuous usage. Therefore, the actual cost of employing computing services will be dependent on our peculiar demands and consumption styles.

3.2 Storage Services

Cloud-based solutions for long-term storage of data for numerous applications and workloads are what we call storage services. Each of these three cloud service providers comes with multiple types of storage solutions such as object, block, backup, and file. The final price includes storage capacity, speed, availability, and redundancy. Table 3.2 shows the monthly rates of various common types of storage in the EU (Ireland) region (as of November 2023).

Storage type	AWS	Azure	Google Cloud
Block storage	€0.075 per GB for gp3 SSD	€0.074 per GB for Premium SSD	€0.124 per GB for pd-balanced SSD

Object storage	€0.018 per GB for S3 Standard	€0.014 per GB for Hot Blob Storage	€0.019 per GB for Standard Storage
File storage	€0.226 per GB for EFS Standard	€0.09 per GB for Files Premium	€0.146 per GB for Filestore Basic
Backup storage	€0.004 per GB for S3 Glacier Deep Archive	€0.001 per GB for Archive Blob Storage	€0.001 per GB for Archive Storage

Table 3.2 Monthly prices for storage services in the EU (Ireland) region

The difference in the cloud providers' storage service pricing is shown in Table 3.2. Prices are almost equal for all storage types except for file storage. As we can see, when it comes to pricing there is only one difference – it turns out that prices in AWS are higher compared to Azure for file storage. Google Cloud charges more than AWS and Azure for block and file storage but less for backup storage.

Storage performance, availability and redundancy affect pricing too. In addition, each of the different storage types provided by AWS and Azure have their pricing set differently also for features such as normal, rarely accessed, and archives. Google Cloud presents you with various options for storage classes ranging from regional, multi-regional, and nearline, depending on your choice of need, and available budget. As a result, the true storage service's price will depend on specific needs and the storage pattern that is adopted in order.

3.4 Network Services

Network services provide connectivity and data transmission for cloud applications and workloads. All three cloud providers charge a fee for data transfer between zones and regions and for outbound traffic to the internet. The price depends on the volume of data, where it comes from and where it goes. The data on transmission costs in Ireland as part of EUROPEAN UNION by November 23 are indicated in Table 3.3.

Data transfer type	AWS	Azure	Google Cloud
Inter-region	€0.02 per GB for Ireland to EU	€0.02 per GB for Ireland to EU	€0.02 per GB for Ireland to EU
Inter-zone	€0.01 per GB for Ireland to UK	€0.01 per GB for Ireland to UK	Free for Ireland to UK
Outbound	€0.087 per GB for first 10 TB	€0.087 per GB for first 10 TB	€0.087 per GB for first 10 TB

Table 3.3 Prices for network services in the EU (Ireland) region

Table 3.3 shows that, except for between zone information transmission, each of the three cloud supplier's charges around no difference for network administrators. Be that as it may, Google Cloud doesn't charge for traffic inside a similar region or zone. Cloud suppliers have many valuing levels for active traffic to the web, subsequently, the rates could change depending upon how much information.

For instance, AWS and Azure offer discounted rates of €0.065 per GB for outbound traffic volumes exceeding 40 TB, while the cost drops to €0.05 per GB for volumes exceeding 100 TB. For expanded outbound traffic volumes, Google Cloud gives limited rates, for example, €0.084 per GB for 140 TB and €0.08 per GB for 400 TB. Subsequently, our singular prerequisites and examples of purpose will decide the genuine expense of organisation administrations.

3.5 Price Conversion Factor Research

- The following procedures were followed to determine the price conversion factor for cloud-based data pipeline tasks based on the average of all data used to convert execution time in seconds to cost:
- Dataflow, Data Factory, and Data Pipeline cloud price information were sourced from online search results.
- Azure Pricing: The Complete Guide - Spot.io, Cloud cost optimisation for data pipelines | Google Cloud Blog, Cost of Cloud Computing: How to Calculate the True Cost of Moving to the Cloud, and Data Pipeline Pricing and FAQ - Data Factory | Microsoft Azure are the four relevant results that were found.
- The cost per execution, cost per GB, cost per hour, and cost per month were collected as price components from each result.
- The following formula was used to convert the price elements to the same unit, which is the cost per second:

$$\text{cost per second} = \frac{\text{cost per unit}}{\text{seconds per unit}}$$

- The average cost per second for each cloud provider was calculated, using the following formula:

$$\text{average cost per second} = \frac{\sum_{i=1}^{n} \text{cost per second}_i}{n}$$

- The average cost per second was used as the price conversion factor to convert the execution time in seconds to cost, using the following formula:

$$\text{cost} = \text{execution time} \times \text{price conversion factor}$$

The results of the price conversion factor finding for data pipeline jobs in the cloud are shown in Table 3.4.

Cloud provider	Average cost per second
AWS	€ 0.00014
Azure	€ 0.00016
Google Cloud	€ 0.00015

Table 3.4 Price conversion factor for data pipeline jobs in the cloud

Table 3.4 shows that there is a little difference in the price conversion factor between the three cloud providers. When compared to Azure, AWS and Google Cloud often have lower price conversion factors, making them the most cost-effective options for cloud-based data pipeline activities. The data pipeline service's kind, location, and consumption rate are additional factors that affect the price conversion factor.

When it comes to Data Pipeline and Data Factory, for instance, you may choose between on-demand, reserved, and serverless pricing models on AWS and Azure, respectively. Dataflow is available on Google Cloud with a variety of pricing models, including batch and streaming. Thus, the real cost of data pipeline services could differ based on your individual requirements and consumption habits.

This research compares the fundamental prices for compute, storage, and network services in the EU (Ireland) area and on the Linux operating system as of November 2023. It does this by analysing AWS, Azure, and Google Cloud. In addition, the study has calculated the price conversion factor for cloud data pipeline operations by averaging all the data, which allows one to translate execution time in seconds to cost.

4 Implementation

This part carries out a detailed examination of the implementation process and discusses the optimization algorithms that are used for optimization. Bat Algorithm (BA) and the original Ant Lion optimiser (OriginalALO) were used for lower execution costs on distributed tasks under multicore environments, mainly with VM's task assignment.

4.1. Bat Algorithm (BA)

The Bat Algorithm is based on the echolocation abilities of bats is a metaheuristic optimization algorithm which is aimed at locating optimum solutions by repeatedly modifying frequency and intensity (Tang, Liu and Pan, 2021). It was invented by Xin-She Yang in 2010 and has risen in popularity for solving different optimization issues. It has the capability of balancing exploration and exploitation properly. Ultrasonic pulses are emitted by bats to detect the location of prey, adjusting the pitch and volume automatically according to the distance.

In addition, it uses a velocity and position vector for each bat, whereby each of these corresponds to an individual solution within the search space. Random walk for exploring and frequency adaptation of exploitation makes the algorithm flexible enough to overcome highly complicated solution areas.

Bat Algorithm has proven useful in jobs like scheduling, function optimization, and most importantly, cloud environments where adaptability makes it appropriate for handling variable and unpredictable conditions. The algorithm's simplicity, versatility, and potential for parallelization contribute to its appeal for solving complex optimization challenges in various domains. In the context of multi-cloud task allocation, BA is configured as follows:

4.1.1 BAT Algorithm Configuration:

- **Population Size (N):** 100 bats.
- **Maximum Iterations:** 500.
- **Loudness (A):** 0.5.
- **Pulse Emission Rate (r):** 0.5.
- **Frequency:** 0.0, 2.0.

4.1.2 Objective Function:

Concerning the objectives for minimizing execution time, it seeks the lowest time of execution from all VMs about the corresponding assigned tasks. It is an iterative algorithm that minimises task allocations as far as possible for execution time.

The results are the minimum execution cost, and the optimized outcome in terms of task allocation to the virtual machines, and the runtime. The optimized solution gives the direction on how the costs are shared between the activities depending on the allocations that minimize the overall execution cost. Its runtime reflects its computational efficiency.

4.2 Original Ant Lion Optimizer (OriginalALO)

The Ant Lion Optimizer is based on the hunting technique of antlion larvae, using a population-based approach (Abualigah *et al.*, 2020). Introduced by Seyedali Mirjalili in the year 2015, ALO imitates both the collaboration as well as the conflict found among Ant and antlion natures. The concept of antlions trapping ants while the ants aim not to fall into these pits is used in this algorithm. In their search, ALO applies this concept by repeatedly repositioning a swarm of virtual ants across a particular search space. It involves two main phases: the random walk phase which sees ants roam around the search space and the trap building phase, where the ants change their positions to the ones associated with fitter solutions.

Similar to the avoidance behaviour of ants against antlion traps, the algorithm creates a competitive component and promotes more diverse searching processes. ALO has proven to be efficient for tackling optimization issues such as functional optimization, classification, and feature extraction (Akinola, O.O., Ezugwu). Because of its ability to respond with flexibility, simplicity and researchers have looked into different designs and hybrids to perfect its performance in various fields. For multi-cloud task allocation, OriginalALO is configured as follows:

4.2.1 Algorithm Configuration:

- Epochs: 100.
- Population Size (pop_size): 50

4.2.3 Problem Definition:

The FloatVar encoding addresses the task allocation problem as a continuous optimization problem. The number of virtual machines determines solution space bounds.

4.2.4 Objective Function:

Using the task assignments, the objective function will calculate the cost of executing the virtual machine with the highest cost. ALO algorithm iteratively modifies the task assignment to minimize the cost.

4.3 Design Specification

4.3.1 Description of the Simulation Environment

This research was developed on the CloudSim simulation environment. CloudSim is an open-source framework for modelling and simulating clouds as a service and infrastructure (Ahmad and Khan, 2019). A scenario was created to mimic multi-cloud deployment that encompassed various datacenters and virtual machines.

Simulation Configuration:

- **Virtual Machines (VMs):** Represents the virtualized computing resources within each datacenter.
- **Number of Datacenters (numDatacenters):** 3 datacenters emulating different cloud providers in Ireland.
- **Number of VMs (numVms):** 8 virtual machines for diverse workload simulation.
- **Number of Cloudlets (numCloudlets):** 4 cloudlets representing tasks or workloads.

4.3.2 Datacenters and Virtual Machines Configuration:

Datacenter Configuration:

- Each datacenter is configured with 2 hosts.

Each host consists of:

- **4 CPU cores:** Simulated using Pe (Processing Element) instances.
- **16GB RAM**: Modelled with RamProvisionerSimple.
- **1TB Storage:** Represented by the host's storage capacity.
- **10Gbps Bandwidth:** Emulated with BwProvisionerSimple.

Datacentre Characteristics:

Data centres are backbone of modern digital operations, functioning as centralized hubs for computing resources and data storage and these facilities are critical for the efficient and secure handling of vast amounts of digital information. Datacentre Characteristics are defined by parameters such as architecture (x86), operating system (Linux), VMM (Xen).

VM Configuration:

The VM configuration helps in performance optimization and with proper configuration the VM operates efficiently, balancing the host system's resources among all running VMs and allocate the right amount of CPU power, memory (RAM), and storage for the VM. VM also shares the physical resources of the host machine with the host OS and other VMs.

- Each VM is created with specifications including:
- 1000 MIPS (Million Instructions Per Second) - is a measure of the computational power of the VM.
- **1 CPU core:** Represented by numberOfPes.
- **512MB RAM:** Modelled with RamProvisionerSimple.
- **10GB Storage:** Represented by the VM's storage capacity.
- **1000 Mbps Bandwidth:** Emulated with BwProvisionerSimple.
- VMs employ a time-shared CloudletScheduler for task execution.

The VM configuration will outline the resources allocated to a VM and some aspects of how it operates, particularly in terms of processing power, memory, storage, bandwidth, and the scheduling of tasks. The use of "simple" provisioners suggests that the allocation is simple, without complex management of resources. The time-shared scheduler implies that multiple tasks can run in parallel, sharing the VM's resources over time.

VM Allocation:

Virtual machines (VMs) are distributed to hosts in datacenters using a random allocation technique. The DatacenterBroker facilitates this process.

Datacentre Broker:

In CloudSim, a Datacentre Broker is an intermediary component that manages the interactions between the end-user and the datacentres. It is responsible for handling the allocation and management of Virtual Machines (VMs), and distribution and execution of computational tasks, or cloudlets. The broker simplifies the user experience by automatically selecting suitable datacentres, provisioning resources, and managing task execution based on predefined criteria such as cost, performance, and resource availability, and helps in efficient utilization of cloud resources in simulations.

4.4 CloudSim Simulation

Task and Cloudlet Creation: The parameters of length, file size, output size, and an appropriate stochastic utilisation model are used to create cloudlets which represent compute tasks. Cloudlets are assigned randomly to associate them with virtual machines according to their tasks in a multi-cloud environment.

Dynamic VM Assignment: During the simulations VMs are randomly assigned to each data center they will be deployed on. Indeed, this dynamic allocation mimics the ever-changing nature of allocating VMs under a multi-cloud environment.

Algorithm Initialization and Configuration: Such parameters include population size, epochs, and solution space bounds that configure the algorithms. Initially, instances of the algorithms are created with a set of parameters. The Bat Algorithm is configured for optimizing task allocation across virtual machines with the goal of minimizing execution costs. It starts with total tasks to be allocated, using a population of 100 bats (each representing a potential solution) and iterating up to 500 times.

Optimization Execution: To provide optimal task allocation solutions and minimize the cost of the solution, the optimization algorithms are performed. To measure each algorithm's computational efficiency, runtime metrics are stored.

5 Results and Findings

In this section, detailed result analysis of utilising the BA and ALO for the optimization of multi-cloud job dissemination is provided. This section offers a detailed analysis of the effects of utilising BA and Original ALO for optimization of multi-cloud task allocation.

5.1 Bat Algorithm Results and Analysis

Minimum Execution Cost:

- The BAT optimisation algorithm resulted in the lowest possible execution cost of €0.02798571429.

Optimised Solution:

The optimised solution, representing task assignments to virtual machines, is.

[1, 1, 1, 2, 0, 3, 1, 0, 2, 1, 3, 2, 3, 3, 1, 0, 0, 2, 2, 2, 2, 0, 3, 2, 2, 1, 2, 1, 2, 1, 3, 2, 1, 2, 0, 0, 2, 2, 0, 1, 0, 0, 2, 2, 1, 2, 2, 1, 3, 3, 2, 2, 1, 1, 2, 3, 2, 2, 3, 3, 0, 1, 0, 3, 2, 1, 2, 1, 3, 2, 2, 2, 2, 3, 2, 0, 2, 2, 0, 1, 2, 3, 0, 3, 0, 1, 3, 1, 2, 3, 1, 0, 1, 3, 1, 1, 0, 1, 1, 2, 3, 2, 0, 1, 3, 0, 2, 2, 3, 3, 2, 2, 2, 2, 2, 3, 0, 1, 3, 3, 2, 3, 2, 1, 1, 2, 2, 2, 0, 3, 3, 2, 2, 1, 2, 2, 2, 2, 0, 2, 0, 1, 1, 1, 1, 0, 1, 2, 2, 0, 3, 1, 1, 1, 2, 0, 0, 0, 2, 2, 1, 3, 0, 1, 3, 0, 2, 1, 1, 1, 3, 1, 2, 3, 3, 2, 1, 3, 2, 1, 1, 1, 1, 2, 0, 1, 0, 3, 3, 3, 3, 1, 3, 0, 0, 2, 3, 1, 2, 0, 2, 1, 1, 1, 2, 1, 3, 3, 0, 3, 2, 2, 0, 3, 2, 2, 1, 3, 3, 1, 1, 1, 1, 1, 2, 2, 1, 2, 3, 1, 1, 2, 0, 2, 1, 1, 0, 0, 2, 0, 3, 0, 1, 0, 2, 2, 2, 3, 2, 1, 1, 0, 2, 0, 2, 3, 0, 2, 1, 3, 1, 2, 1, 3, 2, 2, 3, 1, 1, 1, 1, 3, 0, 2, 2, 3, 2, 2, 2, 2, 3, 2, 1, 2, 2, 3, 2, 2, 2, 2, 2, 2, 3, 2, 0, 2, 2, 2, 2, 2, 0, 0, 1, 2, 2, 3, 2, 3, 1, 3, 1, 1, 2, 1, 0, 2, 1, 1, 1, 1, 2, 3, 1, 1, 2, 0, 0, 3, 2, 3, 0, 1, 2, 0, 3, 0, 1, 2, 2, 1, 3, 1, 2, 1, 2, 0, 3, 0, 1, 2, 1, 2, 2, 1, 1, 3, 2, 0, 2, 2, 2, 2, 3, 3, 1, 2, 0, 1, 2, 3, 1, 3, 2, 2, 2, 2, 1, 1, 2, 3, 2, 2, 2, 1, 3, 3, 2, 1, 2, 3, 2, 2, 2, 2, 1, 0, 2, 1, 3, 1, 3, 1, 1, 1, 1, 2, 2, 3, 1, 2, 2, 2, 2, 2, 1, 2, 3, 0, 2, 1, 3, 1, 2, 3, 2, 1, 0, 2, 1, 3, 1, 1, 1, 0, 1, 2, 3, 2, 1, 2, 3, 2, 0, 0, 2, 3, 2, 1, 3, 2, 2, 0, 1, 2, 1, 1, 1, 0, 0, 3, 3, 2, 2, 1, 2, 0, 2, 1, 3, 1, 3, 3, 1, 2, 1, 2, 0, 2, 2, 1, 3, 2, 2, 3, 1, 3, 1, 1, 1, 0, 1, 2, 2, 2, 2, 1, 3, 3, 3, 2].

Task allocation strategies reveal the effectiveness of the Bat Algorithm in minimizing execution costs.

Runtime Analysis:

- The Bat Algorithm took 7.44 milliseconds to complete the 500 tasks.
- Task allocation strategies reveal the effectiveness of the Bat Algorithm in minimising execution costs.

Algorithmic Insights:

- It is through exploring and exploiting the solution space where BA realises its success in that it adjusts dynamic frequencies and loudness.
- The algorithm gives excellent execution efficiency at the smallest runtime.

5.2 Original Ant Lion Optimizer (OriginalALO) Results and Analysis

Minimum Execution Cost:

A minimal execution cost of €0.02601176471 was obtained using OriginalALO and this value signifies the lowest cost attained through the optimization process.

Optimised Solution:

The optimised solution, representing task assignments to virtual machines, is

[4, 3, 3, 4, 3, 1, 4, 2, 3, 1, 3, 3, 2, 1, 3, 1, 3, 3, 1, 4, 2, 4, 3, 4, 1, 2, 3, 4, 2, 2, 1, 3, 1, 2, 2, 2, 2, 4, 4, 3, 3, 4, 1, 2, 4, 1, 3, 4, 3, 2, 4, 2, 4, 3, 4, 4, 4, 3, 3, 4, 3, 3, 3, 4, 2, 2, 1, 3, 3, 3, 4, 3, 3, 3, 1, 4, 3, 3, 2, 4, 3, 4, 3, 1, 3, 3, 2, 3, 2, 3, 4, 4, 4, 4, 2, 2, 2, 3, 4, 2, 4, 3, 4, 4, 3, 1, 2, 3, 2, 2, 3, 3, 3, 4, 3, 1, 2, 2, 2, 1, 4, 3, 3, 4, 3, 4, 1, 3, 4, 4, 4, 3, 3, 1, 2, 3, 1, 4, 4, 2, 2, 1, 3, 3, 4, 4, 4, 2, 3, 3, 3, 3, 1, 3, 2, 2, 3, 4, 2, 4, 3, 1, 4, 2, 4, 4, 4, 3, 1, 3, 2, 1, 2, 3, 3, 4, 1, 4, 4, 1, 1, 3, 2, 1, 4, 4, 4, 4, 3, 2, 2, 4, 2, 4, 3, 3, 2, 1, 4, 3, 3, 4, 4, 4, 4, 4, 4, 1, 2, 1, 2, 4, 3, 3, 1, 3, 1, 2, 2, 4, 2, 2, 3, 4, 3, 4, 3, 4, 4, 4, 4, 3, 3, 3, 3, 3, 2, 2, 2, 3, 4, 4, 4, 3, 1, 4, 1, 4, 4, 2, 3, 2, 3, 4, 3, 2, 2, 3, 3, 1, 2, 2, 4, 2, 4, 4, 3, 4, 3, 3, 3, 1, 3, 3, 1, 2, 4, 1, 3, 4, 2, 2, 3, 3, 2, 2, 1, 4, 4, 4, 4, 3, 4, 2, 4, 3, 3, 3, 2, 1, 2, 3, 1, 2, 4, 1, 1, 3, 2, 4, 1, 2, 3, 4, 3, 4, 4, 4, 3, 1, 1, 3, 3, 4, 3, 4, 1, 1, 4, 4, 4, 1, 4, 4, 3, 4, 2, 3, 3, 4, 4, 3, 4, 3, 3, 3, 3, 3, 2, 2, 3, 2, 3, 4, 1, 2, 3, 4, 3, 3, 4, 2, 4, 2, 4, 4, 4, 4, 3, 2, 3, 3, 4, 4, 2, 4, 4, 2, 2, 2, 2, 1, 2, 1, 2, 1, 1, 3, 2, 2, 3, 2, 1, 3, 4, 4, 2, 3, 3, 2, 4, 3, 4, 4, 4, 3, 2, 2, 1, 3, 3, 4, 4, 4, 1, 3, 4, 3, 3, 4, 4, 2, 2, 4, 3, 4, 2, 3, 3, 3, 2, 1, 3, 1, 2, 4, 2, 4, 3, 3, 4, 1, 1, 2, 3, 4, 3, 3, 2, 4, 4, 4, 4, 2, 3, 1, 3, 3, 4, 4, 3, 4, 4, 3, 4, 3, 4, 3, 4, 1, 2, 3, 3, 4, 1, 2, 4, 3, 2, 3, 3, 4, 1, 3, 4, 3, 4, 2, 4, 2, 2, 2, 2, 4, 2, 3, 4, 3, 4, 4].

Task allocation strategies highlight the effectiveness of OriginalALO in minimising execution costs.

Runtime Analysis:

- OriginalALO took 219.75 seconds to run the 500 tasks.
- The algorithm demonstrates competitive computational efficiency in converging towards an optimal solution.

Algorithmic Insights:

- ALO adopts antlion's strategy to achieve an appropriate level of balance between exploration and exploitation while at the same time reducing the total cost of execution.
- The algorithm demonstrates efficiency in arriving at almost optimal solutions over a reasonable duration of time.

5.3 Performance Comparison Between BAT and ALO

Minimum Execution Cost Comparison:

Simulated 100, 200 and 500 tasks in this experiment and recorded their respective minimum execution costs. Their results are presented below:

Task Count	BAT	ALO
100	0.00642	0.0054
200	0.011957143	0.010542857
500	0.027985714	0.026011765

Table 5.1 Execution Cost comparison for different number of tasks

Figure 5.1 - Comparison of minimum execution costs between the two algorithms

The above bar plot comparison of the minimum execution cost for BA and OriginalALO for different numbers of tasks. In comparison, the original ALO algorithm provides less minimum cost of executions than the Bat Algorithm.

Runtime Comparison:
For the above simulated experiment, also recorded their respective runtimes. The result is presented below.

Task Count	BAT	ALO
100	7.441821098	41.30995584
200	14.5129931	78.91439891
500	47.88952613	219.7524641

Table 5.2 - Runtime comparison in milliseconds for different number of tasks

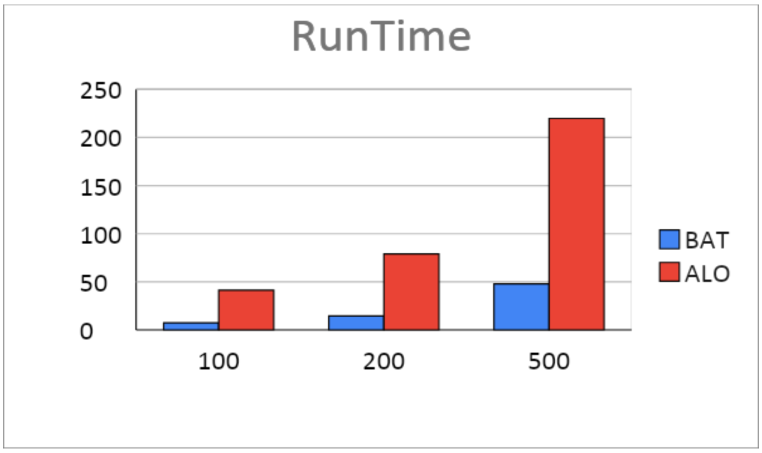

Figure 5.2 Comparison of total runtime between the two algorithms

Above bar plot compares the runtimes of BA and OriginalALO presented in milliseconds for different numbers of tasks. Bat Algorithm is much faster than Original ALO.

Total Execution Cost Comparison in seconds:

Total Execution Cost = Execution Cost * Runtime

Using the above formula, I also calculated the total execution cost for both the algorithms for different numbers of tasks. The result is presented below:

Task Count	BAT	ALO
100	0.047776491	0.223073762
200	0.173533932	0.831983234
500	1.340222596	5.716149389

Table 5.3 - Total execution cost per second comparison in euros for different number of tasks

Figure 5.3 Comparison of total execution cost between the two algorithms

BAT Algorithm gave better results in terms of total execution cost which can be referenced from the above graph.

5.4 Discussion of Findings

Trade-offs Between Cost and Runtime:
The OriginalALO Algorithm has a little lower minimum execution cost than BAT, but it takes a little longer to run. BAT is superior in terms of overall execution cost.

Algorithmic Strengths:
BA excels in efficiently navigating the solution space, demonstrating a dynamic exploration-exploitation balance. OriginalALO exhibits resilience in finding effective solutions through antlion-inspired strategies.

6 Conclusion

The exploitation of the Bat Algorithm (BA) and the Original Ant Lion Optimizer (OriginalALO) in the multi-cloud task allocation has shed light into the suitability and computational efficiency. The comprehensive analysis of results and discussions allows for meaningful conclusions:

Algorithmic Effectiveness:

The Bat Algorithm is characterised by using dynamic exploration and exploitation strategies thus resulting in low execution costs. It appears that the efficiency of the original ant lion optimizer in terms of discovering a near optimal solution for job allocation is simply amazing by considering the behaviour pattern of the antlions.

Trade-offs Between Cost and Runtime:

BAT Algorithm has slightly shorter runtime and achieves slightly better total minimum execution cost than ALO. When choosing an algorithm, decision-makers should look at the trade-offs between computational efficacy and cost-savings.

6.1 Future Work

Bat Algorithm and Original Ant Lion Optimizer still need more fine tuning when solving some multi-cloud problems. Sensitivity analysis and parameter adjustment could reveal optimal setups for different use cases. Investigating hybrid algorithms, which blend the advantages of BA and OriginalALO, may create more powerful and multifaceted optimization programs. Combining different algorithms may therefore allow hybrid approaches to exploit their individual uniqueness for better performance.

The use of hybrid algorithms combining the features of OriginalALO and BA could result in more robust and versatile optimisation methods. Some research projects have employed hybrid techniques that combine different properties of the algorithms to improve performance. To compare the best optimization algorithms, it is necessary to consider how well BA and OriginalALO perform against them. In benchmarks against a wide range of algorithms, it is possible to achieve a better comprehension of the relative advantages and disadvantages of various algorithms.

Using more complex cost models and restrictions can turn algorithms into a more flexible tool for different business cases. The relevant constraints in the optimisation framework may cover issues like integrity of data, privacy, and data compliance to applicable regulations.

References

1. *Jiang, F., Ferriter, K. and Castillo, C., 2020, April. A cloud-agnostic framework to enable cost-aware scheduling of applications in a multi-cloud environment. In NOMS 2020-2020 IEEE/IFIP Network Operations and Management Symposium (pp. 1-9). IEEE.*
2. Abualigah, L. *et al.* (2020) 'Ant Lion Optimizer: A Comprehensive survey of its variants and applications,' *Archives of Computational Methods in Engineering*, 28(3), pp. 1397–1416. https://doi.org/10.1007/s11831-020-09420-6.
3. Wang, Z., Hayat, M.M., Ghani, N. and Shaban, K.B., 2016. Optimizing cloud-service performance: Efficient resource provisioning via optimal workload allocation. IEEE Transactions on parallel and Distributed Systems, 28(6), pp.1689-1702.
4. Ahmad, M.O. and Khan, R.Z. (2019) 'Cloud Computing Modeling and Simulation using CloudSim Environment,' *International Journal of Recent Technology and Engineering*, 8(2), pp. 5439–5445. https://doi.org/10.35940/ijrte.b3669.078219.
5. Chatzithanasis, G. *et al.* (2021b) 'Exploring cost-efficient bundling in a multi-cloud environment,' *Simulation Modelling Practice and Theory*, 111, p. 102338. https://doi.org/10.1016/j.simpat.2021.102338.
6. Chimakurthi, V.N.S.S. (2020) 'The challenge of achieving zero trust remote access in Multi-Cloud environment,' *ABC Journal of Advanced Research*, 9(2), pp. 89–102. https://doi.org/10.18034/abcjar.v9i2.608.
7. Hong, J. *et al.* (2019b) 'An overview of multi-cloud computing,' in *Advances in intelligent systems and computing*, pp. 1055–1068. https://doi.org/10.1007/978-3-030-15035-8_103.
8. Raj, P. and Surianarayanan, C. (2021b) 'Multi-Cloud adoption challenges for the Cloud-Native Era,' *International Journal of Cloud Applications and Computing*, 11(2), pp. 67–96. https://doi.org/10.4018/ijcac.2021040105.
9. Tang, J., Liu, G. and Pan, Q. (2021) 'A review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends,' *IEEE/CAA Journal of Automatica Sinica*, 8(10), pp. 1627–1643. https://doi.org/10.1109/jas.2021.1004129.
10. Akinola, O.O., Ezugwu, A.E., Agushaka, J.O., Zitar, R.A. and Abualigah, L., 2022. Multiclass feature selection with metaheuristic optimization algorithms: a review. Neural Computing and Applications, 34(22), pp.19751-19790.

11. Yeganeh, B. *et al.* (2020) 'A first comparative characterization of multi-cloud connectivity in today's internet,' in *Lecture Notes in Computer Science*, pp. 193–210. https://doi.org/10.1007/978-3-030-44081-7_12.
12. Jayakumar, V.K., Lee, J., Kim, I.K. and Wang, W., 2020, May. A self-optimized generic workload prediction framework for cloud computing. In 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS) (pp. 779-788). IEEE.
13. Casola, V., Benedictis, A.D., Rak, M., Villano, U., Rios, E., Rego, A. and Capone, G., 2019. Model-based deployment of secure multi-cloud applications. International Journal of Grid and Utility Computing, 10(6), pp.639-653.
14. Imran, H.A., Latif, U., Ikram, A.A., Ehsan, M., Ikram, A.J., Khan, W.A. and Wazir, S., 2020, November. Multi-cloud: a comprehensive review. In 2020 IEEE 23rd International Multitopic Conference (INMIC) (pp. 1-5). IEEE.
15. Al-Roomi, M., Al-Ebrahim, S., Buqrais, S. and Ahmad, I., 2013. Cloud computing pricing models: a survey. International Journal of Grid and Distributed Computing, 6(5), pp.93-106.
16. Yang, X.S., 2010. A new metaheuristic bat-inspired algorithm. In Nature inspired cooperative strategies for optimization (NICSO 2010) (pp. 65-74). Berlin, Heidelberg: Springer Berlin Heidelberg.
17. Mirjalili, S., 2015. The ant lion optimizer. Advances in engineering software, 83, pp.80-98.