International Journal on Cybernetics & Informatics (IJCI) Vol.12, No.6, December 2023

ESTIMATING CLOUD COMPUTING ROUND-TRIP TIME (RTT) USING FUZZY LOGIC FOR INTER-REGION DISTANCES

Ihab Sekhi

Institute of Information Technology, Miskolc University, 3515 Miskolc

ABSTRACT

Cloud computing is widely considered a transformative force in the computing world and is poised to replace the traditional office setup as an industry standard. However, given the relative novelty of these services and challenges such as the impact of physical distance on round-trip time (rtt), questions have arisen regarding system performance and associated billing structures. The primary objective of this study is to address these concerns. We aim to alleviate doubts by leveraging a fuzzy logic system to classify distances between regions that support computing services and compare them with the conventional web hosting format. To achieve this, we analyse the responses of one of these services, like amazon web services, across different distance categories (near, medium, and far) between regions and strive to conclude overall system performance. Our tests reveal that significant data is consistently lost during customer transmission despite exhibiting superior round-trip times. We delve into this issue and present our findings, which may illuminate the observed anomalous behaviour.

KEYWORDS

Round Trip Time, Wireless Network, SLA, cloud computing.

1. INTRODUCTION

The term "cloud" is employed metaphorically to represent the internet and serves as a simplified representation of the intricate infrastructure it masks. Cloud computing is a sweeping concept that encompasses contemporary and widely recognised technological trends, including software as a service (SaaS), network as a service (NaaS), and everything as a service (XaaS). These innovations share a common theme of depending on the internet to meet the computing requirements of users. One such element is SaaS, which stands for Software as a Service [1].

The application operates exclusively in the cloud, utilizing the servers provided by the service provider via the Internet. The client is a straightforward web browser or a similarly uncomplicated client. For instance, Amazon Web Services offers popular business applications accessible online through a web browser, with the software and data stored on Amazon's servers. The potential of cloud computing is unquestionable when implemented effectively.

There are speculations that it could herald a new era in computing, possibly becoming the industry norm [2].

Similar to how only a few individuals today choose to construct their own houses but instead opt to rent one, in the upcoming generation of computing, people may favour selecting a scalable and dependable provider for their computing requirements. This choice can significantly reduce risks when launching a new application instead of building an entirely new enterprise solely for

Dhinaharan Nagamalai (Eds): NLAII, VISCOM, CCSITA, AIDD, BCYIoT -2023
pp. 95-105, 2023. IJCI – 2023 DOI:10.5121/ijci.2023.120608

product launches. Our motivation for conducting this measurement study is driven by the extensive hype surrounding the concept of cloud computing. Despite the extensive discussions surrounding network availability in the cloud environment, such as issues of reliability and latency, there is a noticeable absence of empirical measurement studies to validate these claims. Additionally, no explicit comparisons have been drawn between networking performance metrics, such as Round-Trip Time (RTT), and the actual RTT experienced by a web hosting service across different geographical regions.

As a result, our research endeavours to assess the performance of networking services under varying load conditions to determine the validity of the hype generated around cloud computing. We approach the assessment of network availability from two broad perspectives: firstly, by computing network-based RTT through ping tests to evaluate connectivity, and secondly, by adopting a mathematical approach to verify the scalability and performance claims made by cloud service providers [3].

To better understand these aspects, we employ a fuzzy logic system featuring three triangular membership functions for two input parameters: distance and time. This system allows us to measure the performance and scalability regarding the expected optimal RTT. Our study focuses on the Amazon Web Services platform, where we measure the performance when retrieving images categorized as small (RTT < 100 ms), medium (100 ms < RTT < 180 ms), and large (RTT > 180 ms). Subsequently, we conduct a comparative analysis of these findings. The comprehensive implementation details and the evaluation of our results will be elaborated upon later in this paper.

The structure of this paper is as follows:

- Section 2 provides an overview of related work, with an emphasis on network aspects, parameters, and diverse trust models.
- Section 3 introduces the framework for our cloud network methodology, specifically focusing on RTT.
- Section 4 details the implementation of our work.
- Section 5 conducts an evaluation of our work and compares the results of the RTT calculations.
- Finally, the paper concludes and summarizes the findings in the last section.

2. RELATED WORK

Cloud computing represents a relatively recent and evolving concept, with the current array of services being in their early stages of development. Consequently, there exists a dearth of scholarly literature pertaining to this particular domain. Furthermore, the absence of established industry standards has led each service provider to define its parameters for resource utilization. Measurements in the realm of cloud computing services can be broadly categorized into two groups: computation-based measurements and pamphlet work-based measurements. Our research aligns with the latter category.

Complementing our investigation is the realm of computation-based measurements, which encompasses factors like Round Trip Time (RTT) for networking performance. Such measurements require direct access to the service provider's servers, thus necessitating involvement from the servers themselves or authorized third parties, including individual users or enterprises that maintain affiliations with cloud service providers. These authorized third-party measurement services play a crucial role in monitoring network latency performance and reporting the system's overall health, particularly for cloud platforms like Amazon's EC2.

3. METHODOLOGY

In this section, we provide a summary of the existing status of cloud computing systems, outline the types of measurement tests applicable to these systems, and elucidate the methodologies employed in structuring our testing model.

3.1. The Contemporary Status of Cloud Computing Services

In today's market, a multitude of cloud computing services are accessible, each offering a wide range of services. These offerings encompass potent tools like Amazon Web Services, more specialized options [4], and the comprehensive server solution delivered by Google App Engine [5]. Amazon was at the forefront of bringing cloud computing services to the general public, and it still offers a comprehensive range of instance types tailored to suit various use cases. These instance types are composed of diverse CPU, memory, storage, and networking capacity combinations, allowing you to select the ideal resource configuration for your applications. Each instance type comes in one or more instance sizes, enabling you to adjust your resources easily to match the demands of your specific workload. Storing data is facilitated through the Simple Storage Service (S3), allowing objects of up to 5GB to be stored.

Additionally, Amazon has developed a limited database layer atop S3. The Simple Queue Service (SQS) serves as a message-passing API to enable communication among deployed AMIs. Most tasks within the Amazon cloud are executed through command line interfaces, with Amazon offering a variety of tools, including robust security options, for issuing commands to AMI collections.

Currently, numerous cloud computing services offer features comparable to Amazon EC2. Among these competitors is Mosso [6], GoGrid and AppNexus. Google App Engine and Amazon Web Services (AWS) offer contrasting services. While AWS's Elastic Compute Cloud (EC2) provides root privileges and allows various organizations to purchase on-demand computing services through a pay-as-you-go model, Google App Engine focuses on cloud-based web services with tailored frameworks and specific building blocks [7].

Amazon's EC2 grants users internet-based access to a virtual pool of computers, essentially giving them the capabilities of owning high-quality PC hardware, including CPU, GPU, RAM, storage options, and a choice of operating systems. Additionally, AWS offers pre-configured application programming, such as CRM, databases, and web hosting servers [8].

Amazon also maintains a global backup system for AWS servers [9]. The pricing of AWS services depends on factors like the selected tools, operating systems, program design, accessibility needs, security requirements, and administrative tools. Clients have the flexibility to purchase dedicated virtual AWS instances, physical machines, or a combination of both. Security is a fundamental aspect of Amazon's service agreements, and AWS operates across multiple geographic regions [10].

In 2017, AWS provided approximately 90 services encompassing database management, storage, application services, hardware, IoT devices, and monitoring [11]. AWS's primary offerings include AS3 and EC2, with most management tasks accessible through APIs for seamless application integration. AWS supports HTTP and employs REST and SOAP protocols [12].

Amazon markets AWS as a faster and cost-effective solution for scaling computations compared to traditional server farms [13]. While AWS dominates the cloud computing market, Microsoft and Google compete [14].

3.2. Measurement Tests

Cloud computing system tests can be broadly divided into two primary categories: evaluations centered around resource assessment and assessments focused on network infrastructure. The first category involves evaluating the actual computational performance of the machines used to operate cloud applications. Some tests, like those related to storage capacity and memory utilization, follow established standards. However, it's crucial to recognize that each cloud provider defines its criteria and methods for assessing CPU utilization.

For example, Google App Engine quantifies CPU utilization in terms of "Megacycles used," which some users might see as less intuitive. In contrast, Amazon EC2 measures CPU utilization by considering the deployment duration of a machine instance and the number of instances used. Access to conduct these tests typically requires root access to the server, making them primarily the domain of the cloud provider or authorized third parties.

An example of such an authorized third party is Hypericin Inc., which continuously monitors the performance of both EC2 and App Engine in real-time and shares the results on its website, Cloud Status [6].

The primary focus of our research revolves around assessing the network performance for requests handled by cloud-deployed applications. This area involves various measurements, including the round-trip time (RTT). The Round-Trip Time (RTT) signifies the duration between sending a message from a source to a remote location and its subsequent return to the source. The significance of this metric becomes clear as it precisely quantifies the latency users encounter when interacting with a web service by timing the interval between the submission of a query and the receipt of the corresponding response.

3.3. Testing Model

The Our approach to assessing network performance in cloud computing services begins with the execution of a ping network test method, establishing connections between all regions within AWS. Subsequently, we validate the results by applying a mathematical equation to calculate the average round-trip time (RTT). It is important to note that we assume the application's performance to be consistent in both scenarios as we compare the system's duration to the transfer times across the network.

Following this, a fuzzy logic system is employed, comprising three membership functions, to categorize the proximity between regions into three groups: "near," "average," or "far." This categorization serves as the primary input parameter for the fuzzy inference method. Furthermore, three membership functions are used to classify time as either "short," "medium," or "long," serving as the secondary input for the fuzzy logic system.

To sum up, a comparative study is carried out on the two results obtained for round-trip time.

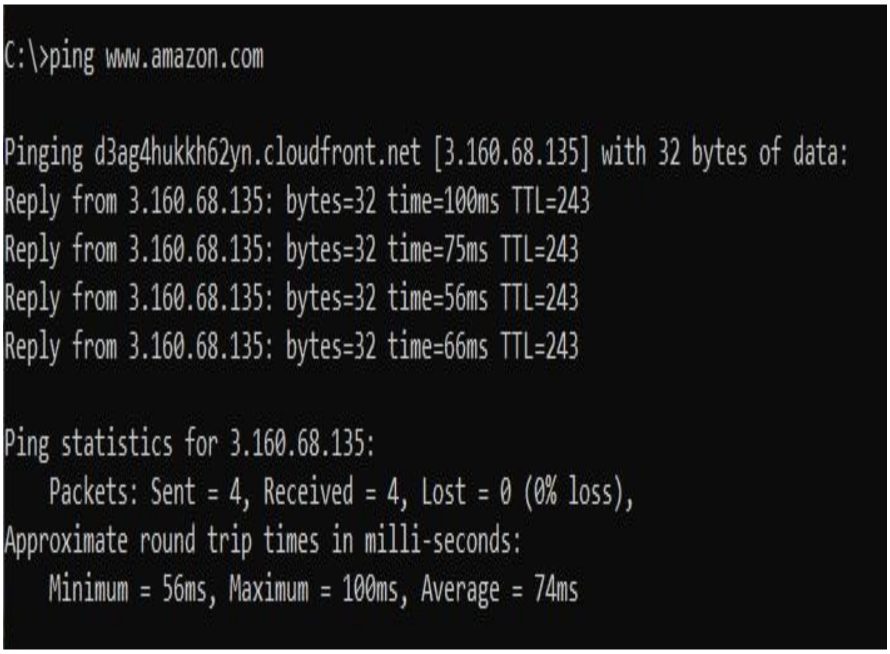
The fuzzy logic method is suitable for estimating and approximating RTT within cloud computing.

4. INFRASTRUCTURE APPLICATION

This part provides a detailed explanation of the steps that must be taken to put the test model described in the previous section into action.

4.1. The Initial Technique

The Ping Test serves as a quick and precise tool for assessing the quality of your internet connection. It measures the millisecond delay between your computer and the chosen remote server. The distance to the server significantly influences the ping value - the greater the distance, the higher the ping value. A chart resembling a consistently straight horizontal line indicates a stable connection [15]. The figure1 presented provides a sample of our work, demonstrating the calculation of Round-Trip Time (RTT) for www.amazon.com, while the remaining results are available in the accompanying working model.



```
C:\>ping www.amazon.com

Pinging d3ag4hukkh62yn.cloudfront.net [3.160.68.135] with 32 bytes of data:
Reply from 3.160.68.135: bytes=32 time=100ms TTL=243
Reply from 3.160.68.135: bytes=32 time=75ms TTL=243
Reply from 3.160.68.135: bytes=32 time=56ms TTL=243
Reply from 3.160.68.135: bytes=32 time=66ms TTL=243

Ping statistics for 3.160.68.135:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 56ms, Maximum = 100ms, Average = 74ms
```

Figure 1. Ping testing sample

4.2. The Second Technique

To calculate the round-trip time using a mathematical method, let us consider a typical network topology where an Ex inda appliance is strategically positioned between the client and the server.

When each packet traverses through the Exinda appliance, it is meticulously time-stamped using a highly precise nanosecond resolution clock source. Given that the Exinda appliance intercepts the packet after it has been sent by the client, the initial transmission time is unknown. Consequently, the round-trip time (RTT) is determined by adding together the round-trip time from the appliance to the server and back (Server RTT), as well as the round-trip time from the appliance to the client and back (Client RTT). As more packets are dispatched from the client, passing through the Exinda appliance to complete the round trip, the RTT estimate is continuously refined by averaging the newly acquired information [16].

The ensuing diagram and equations provide a visual representation of how the round-trip time is computed:

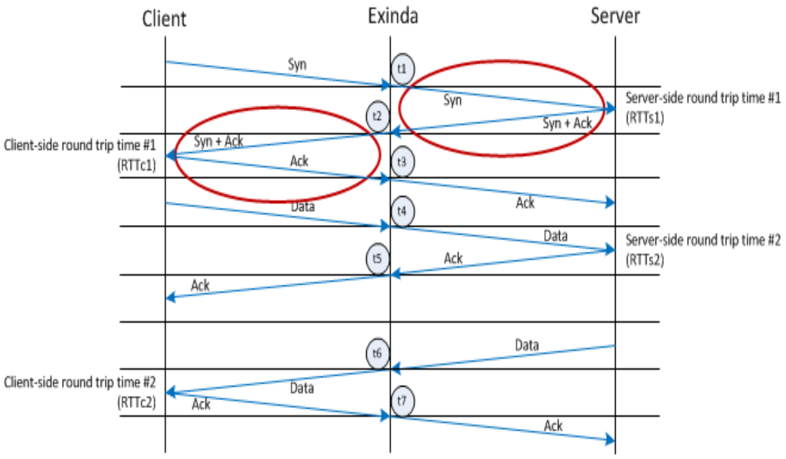


Figure 2. RTT calculated

Server RTT:

- RTTs1 = t2 - t1
- RTTs2 = t5 - t4

Client RTT:

- RTTc1 = t3 - t2
- RTTc2 = t7 - t6

Average RTT:
Average Server RTT = (RTTs1 + RTTs2)/2
Average Client RTT = (RTTc1 + RTTc2)/2
Average Total RTT = avRTTs + avRTTc

4.3. Fuzzy Logic System

4.3.1. Fuzzification

Fuzzy logic involves mapping a dataset to scalar data as output. This system comprises four main components: fuzzification, inference rules, decision components, and defuzzification. Figure 3 illustrates the components of the fuzzy system.

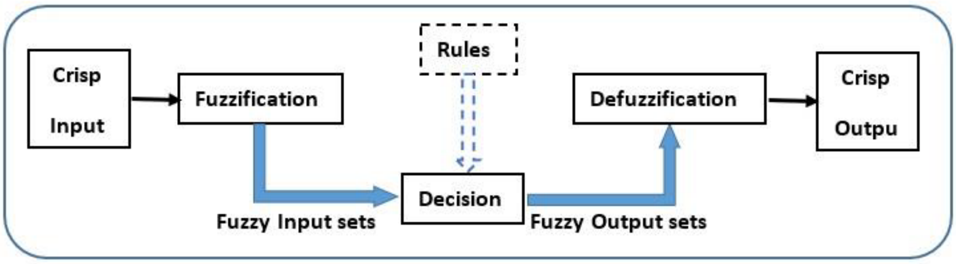


Figure 3. Fuzzification Process

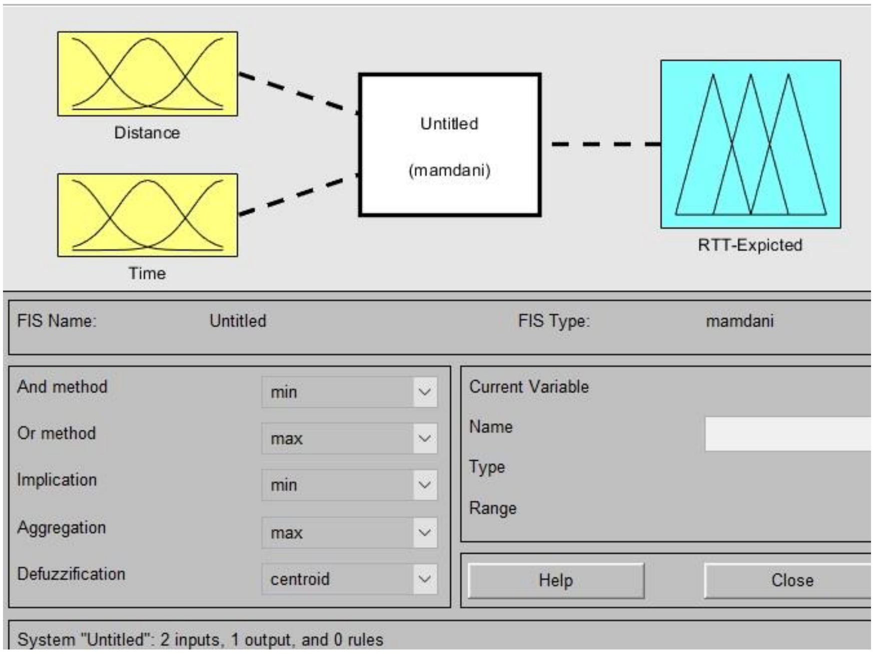
The fuzzification system takes crisp values as input and transforms them into fuzzy logic sets using linguistic set variables, terms, and fuzzy membership functions. This process is known as fuzzification. Subsequently, fuzzy inference rules are applied to obtain the fuzzy outcome value. The final step is defuzzification, which returns the fuzzy outcome to a crisp value [17].

4.3.2. Fuzzy Input

The proposed model employs a triangular membership function [18], represented by equation (1), to convert crisp values to fuzzy sets. This function involves a vector "d" and relies on three scalar arguments: l, m, and n. The values for distance and time are calculated using a fuzzy logic system, and the membership function graph (equation 1) is plotted.

$$Triangled(d\colon l,m,n)=\begin{cases} 0,\ d<l \\ d-l/m-l,\ l\le d\le m \\ n-d/n-m,\ m\le d\le n \\ 0,\ n\le d \end{cases} \tag{1}$$

The subsequent illustrations depict the inputs and outputs of the fuzzy logic system, respectively.



FIS Name: Untitled

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

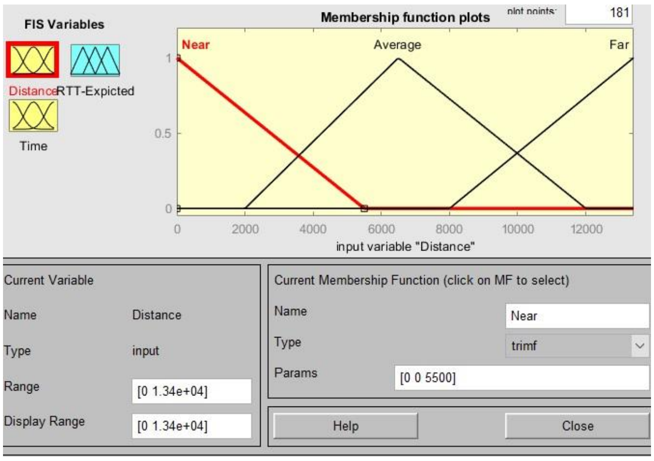
Name

Type

Range

System "Untitled": 2 inputs, 1 output, and 0 rules

Figure 4. Fuzzy logic designer



Current Variable

Name Distance

Type input

Range [0 1.34e+04]

Display Range [0 1.34e+04]

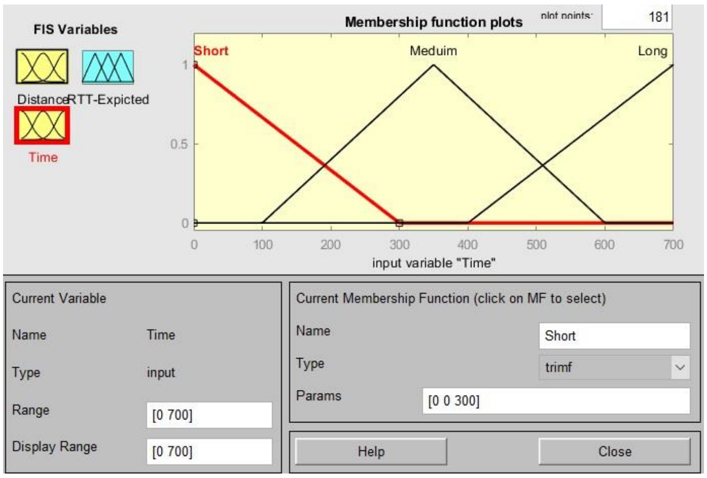
Current Membership Function (click on MF to select)

Name Near

Type trimf

Params [0 0 5500]

Figure 5. Fuzzy input (Distance)



Current Variable

Name Time

Type input

Range [0 700]

Display Range [0 700]

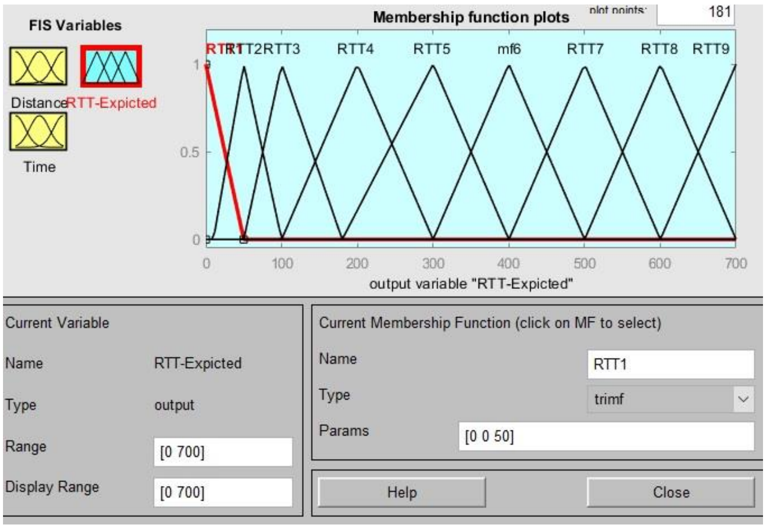
Current Membership Function (click on MF to select)

Name Short

Type trimf

Params [0 0 300]

Figure 6. Fuzzification Process



Current Variable

Name RTT-Expicted

Type output

Range [0 700]

Display Range [0 700]

Current Membership Function (click on MF to select)

Name RTT1

Type trimf

Params [0 0 50]

Figure 7. Defuzzification Process

5. EVALUATION

5.1. Figures and Tables

Table 1. Final compare result.

Region	**RTT(Standard)**	**RTT (Fuzzy logic)**
Africa (Cape Town)	554	487
Asia Pacific (Hong Kong)	388	293
Asia Pacific (Tokyo)	640	421
Asia Pacific (Seoul)	433	308
Asia Pacific (Osaka-Local)	557	377
Asia Pacific (Mumbai)	126	117
Asia Pacific (Hyderabad)	370	234
Asia Pacific (Singapore)	365	293
Asia Pacific (Sydney)	640	620
Asia Pacific (Jakarta)	411	298
Asia Pacific (Melbourne)	640	619
Canada (Central)	351	332
EU (Frankfurt)	562	262
Europe (Zurich)	560	256
Europe (Stockholm)	536	227
Europe (Milan)	640	266
Europe (Spain)	605	533
EU (Ireland)	325	265
EU (London)	261	247
Europe (Paris)	489	259
Middle East (UAE)	91	16.1
Middle East (Bahrain)	100	16.2
South America (São Paulo)	640	592
US East (N. Virginia)	301	298
US East (Ohio)	640	501
US West (N. California)	640	489
US West (Oregon)	386	366

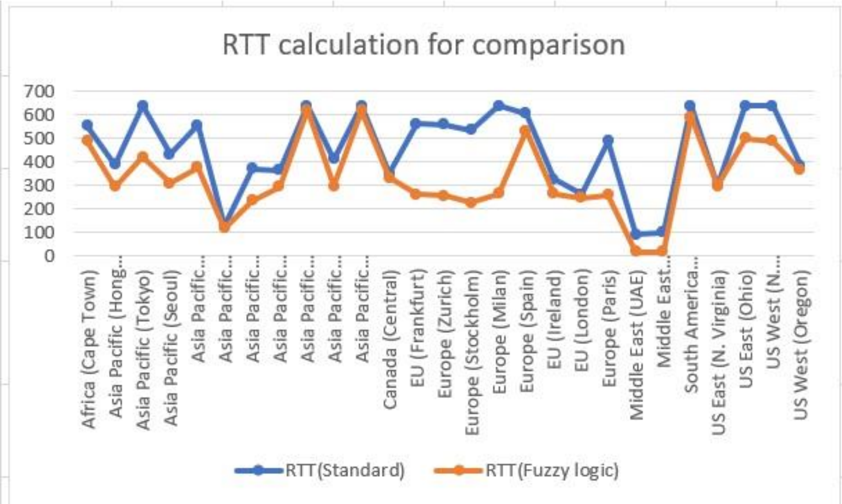


Figure 8. RTT calculation for comparison

6. CONCLUSIONS

In this paper, we introduce an innovative model for assessing network Round-Trip Time (RTT) in the context of the cloud service provider AWS, employing a fuzzy logic mechanism. Our model demonstrates enhanced efficiency when contrasted with conventional RTT calculation models, as it leverages fuzzy logic to offer detailed and descriptive results. Furthermore, it exhibits adaptability for deployment in systems of varying scales. This model empowers users by providing valuable insights for making informed choices regarding precise and fitting Service Level Agreement (SLA) guarantees from cloud resource providers.

REFERENCES

[1] Godse, m., & mulik, s. (2009, september). An approach for selecting software-as-a-service (saas) product. In 2009 ieee international conference on cloud computing (pp. 155-158). Ieee.

[2] Buyya, r., yeo, c. S., & venugopal, s. (2008, september). Market-oriented cloud computing: vision, hype, and reality for delivering it services as computing utilities. In 2008 10th ieee international conference on high performance computing and communications (pp. 5-13). Ieee.

[3] Padmanabhan, v. N., & subramanian, l. (2001, august). An investigation of geographic mapping techniques for internet hosts. In proceedings of the 2001 conference on applications, technologies, architectures, and protocols for computer communications (pp. 173-185).

[4] Saraswat, m., & tripathi, r. C. (2020, december). Cloud computing: comparison and analysis of cloud service providers-aws, microsoft and google. In 2020 9th international conference system modeling and advancement in research trends (smart) (pp. 281-285). Ieee.

[5] Zahariev, a. (2009). Google app engine. Helsinki university of technology, 1-5.

[6] Jin, h., ibrahim, s., bell, t., gao, w., huang, d., & wu, s. (2010). Cloud types and services. Handbook of cloud computing, 335-355.

[7] Mitra, a., o'regan, n., & sarpong, d. (2018). Cloud resource adaptation: a resource based perspective on value creation for corporate growth. Technological forecasting and social change, 130, 28-38.

[8] Díaz, m., martín, c., & rubio, b. (2016). State-of-the-art, challenges, and open issues in the integration of internet of things and cloud computing. Journal of network and computer applications, 67, 99-117.

[9] Vrable, m., savage, s., & voelker, g. M. (2012, february). Bluesky: a cloud-backed file system for the enterprise. In fast (vol. 12, pp. 1-14).

[10] Coppolino, l., d’antonio, s., mazzeo, g., & romano, l. (2017). Cloud security: emerging threats and current solutions. Computers & electrical engineering, 59, 126-140.

[11] Marinescu, d. C. (2022). Cloud computing: theory and practice. Morgan kaufmann.

[12] Sultan, n. (2015). The implications of cloud computing for global enterprise management. In global enterprise management: new perspectives on challenges and future developments volume ii (pp. 39-56). New york : palgrave macmillan us.

[13] Galloway, s. (2018). The four: the hidden dna of amazon, apple, facebook, and google. Penguin.

[14] Gajbhiye, a., & shrivastva, k. M. P. (2014, september). Cloud computing: need, enabling technology, architecture, advantages and challenges. In 2014 5th international conference-confluence the next generation information technology summit (confluence) (pp. 1-7). Ieee.

[15] Goel, u., wittie, m. P., claffy, k. C., & le, a. (2015). Survey of end-to-end mobile network measurement testbeds, tools, and services. Ieee communications surveys & tutorials, 18(1), 105-123.

[16] Mirkovic, d., armitage, g., & branch, p. (2018). A survey of round trip time prediction systems. Ieee communications surveys & tutorials, 20(3), 1758-1776.

[17] Klir, g. J., & yuan, b. (1996). Fuzzy sets and fuzzy logic: theory and applications. Possibility theory versus probab. Theory, 32(2), 207-208.https://doi.10.5860/choice.33-2786

[18] Pedrycz, w. (1994). Why triangular membership functions ?. Fuzzy sets and systems, 64(1), 21-30. Https://doi.org/10.1016/0165-0114(94)90003-5

AUTHORS

Ihab Razzaq Sekhi. From Iraq. An employee in the Iraqi Ministry of Education – Southern Technical University. And a graduate student in the doctoral stage. PhD student in Hungary - Miskolc University.