



Universidade Federal  
de Ouro Preto

**Universidade Federal de Ouro Preto  
Instituto de Ciências Exatas e Aplicadas  
Departamento de Computação e Sistemas**

**Marcos Paulo de Souza Costa**

**Análise da Computação em Nuvem  
Aplicada à Indústria 4.0**

**João Monlevade  
2023**

**Marcos Paulo de Souza Costa**

# **Análise da Computação em Nuvem Aplicada à Indústria 4.0**

Monografia apresentada ao curso de Engenharia de Computação do Instituto de Ciências Exatas e Aplicadas, da Universidade Federal de Ouro Preto, como requisito parcial para aprovação na Disciplina “Trabalho de Conclusão de Curso II”.

Orientador: Theo Silva Lins.

**João Monlevade**

**2023**



## FOLHA DE APROVAÇÃO

**Marcos Paulo de Souza Costa**

### **Análise da Computação em Nuvem Aplicada à Indústria 4.0**

Monografia apresentada ao Curso de Engenharia de Computação da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Bacharelado em Engenharia de Computação

Aprovada em 30 de março de 2023

#### Membros da banca

Doutor - Theo Silva Lins - Orientador(a) Universidade Federal de Ouro Preto  
Doutor - Marlon Paolo Lima - Universidade Federal de Ouro Preto  
Doutor - Filipe Nunes Ribeiro - Universidade Federal de Ouro Preto

Theo Silva Lins, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 07/04/2023



Documento assinado eletronicamente por **Theo Silva Lins, PROFESSOR DE MAGISTERIO SUPERIOR**, em 10/04/2023, às 18:58, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [http://sei.ufop.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **0506326** e o código CRC **F9A7C9B9**.

*Dedico este trabalho à cada sorriso que recebi durante essa trajetória.*

# Agradecimentos

Agradeço aos meus amigos e familiares que estiveram ao meu lado durante toda a minha trajetória acadêmica me apoiando e impulsionando sempre a ser uma pessoa melhor.

*“São as nossas escolhas que revelam o que realmente somos, muito mais do que as nossas qualidades.”*

— J.K. Rowling,  
*em: Harry Potter e a Câmara Secreta.*

# Resumo

A cada instante surge uma nova tecnologia com o intuito de ajudar e facilitar a vida das pessoas. A indústria é um dos grandes beneficiados pela crescente atualização dessas novidades fazendo com que seus processos sejam cada vez mais automatizados e padronizados e menos dependentes da mão de obra humana. Com a expansão da Indústria 4.0 vem crescendo o interesse em seus principais pilares, em especial a Computação em Nuvem, que é responsável pelo armazenamento e processamento de dados em grande parte do processo industrial. A Computação em Nuvem tem se tornado cada vez mais a melhor opção para as empresas, possibilitando a elas a terceirização de hardware, manutenção e serviços que elas próprias deveriam dispor. O propósito deste trabalho é testar diferentes modelos de nuvens disponíveis no mercado e então avaliá-los e classificá-los com base em suas vantagens e desvantagens na aplicação à Indústria 4.0.

**Palavras-chaves:** Computação em Nuvem. Indústria 4.0.

# Abstract

Every moment a new technology comes up to help and make life easier for people. The industry is one of the biggest benefited by the increasing updating of these news making their processes increasingly automated and standardized and less dependent on human labor. With the expansion of Industry 4.0, interest has grown in its main pillars, especially Cloud Computing, which is responsible for storing and processing data in much of the industrial process. Cloud computing is increasingly becoming the best choice for companies, by enabling them to outsource the hardware, maintenance and services they should have themselves. The purpose of this paper is to test different cloud models available in the market and then to evaluate and classify them based on their advantages and disadvantages when applying to Industry 4.0.

Keywords: Cloud Computing. Industry 4.0.



# Lista de ilustrações

Figura 1 – Braço robótico ED-7220C Fonte: (RODRIGUES, 2022) . . . . .	22
Figura 2 – Graus de liberdade do braço robótico Fonte: (RODRIGUES, 2022) . .	23
Figura 3 – Fotografia do espaço de trabalho Fonte: (RODRIGUES, 2022) . . . . .	28
Figura 4 – Diagrama de rede local Fonte: elaborado pelo autor . . . . .	30
Figura 5 – Execução na conexão local Fonte: elaborado pelo autor . . . . .	31
Figura 6 – Diagrama da nuvem privada Fonte: elaborado pelo autor . . . . .	32
Figura 7 – Ativação do OpenNebula Fonte: elaborado pelo autor . . . . .	33
Figura 8 – Acesso ao OpenNebula Fonte: elaborado pelo autor . . . . .	33
Figura 9 – Interface do OpenNebula Fonte: elaborado pelo autor . . . . .	34
Figura 10 – Diagrama da nuvem pública Fonte: elaborado pelo autor . . . . .	34
Figura 11 – Interface de Instâncias da VM <i>Google Cloud</i> Fonte: elaborado pelo autor	35
Figura 12 – Servidor aguardando conexão do cliente Fonte: elaborado pelo autor . .	36
Figura 13 – Cliente conectado à <i>Google Cloud</i> Fonte: elaborado pelo autor . . . . .	36
Figura 14 – Diagrama da nuvem híbrida Fonte: elaborado pelo autor . . . . .	37
Figura 15 – Servidor conectado através do Putty Fonte: elaborado pelo autor . . .	38
Figura 16 – Conexão cliente e servidor nuvem híbrida Fonte: elaborado pelo autor .	39
Figura 17 – Variação das latências entre as conexões das 10 execuções na conexão local Fonte: elaborado pelo autor . . . . .	42
Figura 18 – Valor médio das latências na conexão local Fonte: elaborado pelo autor	43
Figura 19 – Variação das latências entre as conexões das 10 execuções na nuvem privada Fonte: elaborado pelo autor . . . . .	45
Figura 20 – Valor médio das latências na nuvem privada Fonte: elaborado pelo autor	46
Figura 21 – Variação das latências entre as conexões das 10 execuções na nuvem pública Fonte: elaborado pelo autor . . . . .	48
Figura 22 – Valor médio das latências na nuvem pública Fonte: elaborado pelo autor	49
Figura 23 – Variação das latências entre as conexões das 10 execuções na nuvem híbrida Fonte: elaborado pelo autor . . . . .	51
Figura 24 – Valor médio das latências na nuvem híbrida Fonte: elaborado pelo autor	52
Figura 25 – Tempo médio de execução por configuração Fonte: elaborado pelo autor	53
Figura 26 – Latência média da conexão por configuração Fonte: elaborado pelo autor	54

# Lista de tabelas

Tabela 1 – ED-7220C: Características Fonte: (RODRIGUES, 2022) . . . . .	23
Tabela 2 – Tempo total de execução do código cliente na conexão local . . . . .	40
Tabela 3 – Latência total de execução do código cliente na conexão local . . . . .	41
Tabela 4 – Tempo total de execução do código cliente na nuvem privada . . . . .	44
Tabela 5 – Latência total de execução do código cliente na nuvem privada . . . . .	44
Tabela 6 – Tempo total de execução do código cliente na nuvem pública . . . . .	46
Tabela 7 – Latência total de execução do código cliente na nuvem pública . . . . .	47
Tabela 8 – Tempo total de execução do código cliente na nuvem híbrida . . . . .	50
Tabela 9 – Latência total de execução do código cliente na nuvem híbrida . . . . .	50
Tabela 10 – Média tempo de execução . . . . .	52
Tabela 11 – Média da latência por conexão . . . . .	54
Tabela 12 – Média da latência total . . . . .	55

# Lista de abreviaturas e siglas

CPS	<i>Cyber-Physical Systems</i> (Sistemas Ciber-Físicos)
IDE	<i>Integrated Development Environment</i> (Ambiente de desenvolvimento integrado)
IaaS	<i>Infrastructure as a Service</i> (Infraestrutura como Serviço)
IoT	<i>Internet of Things</i> (Internet das Coisas)
PAAS	<i>Platform as a Service</i> (Plataforma como Serviço)
RA	Realidade Aumentada
RTT	Round-Trip Time
SaaS	<i>Software as a Service</i> (Software como Serviço)
SSH	<i>Secure Shell</i>
TCP	<i>Transmission Control Protocol</i>
TI	Tecnologia da Informação
UDP	<i>User Datagram Protocol</i>
VPN	<i>Virtual Private Network</i> (Rede Privada Virtual)

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
<b>1.1</b>	<b>Problema</b>	<b>16</b>
<b>1.2</b>	<b>Justificativa</b>	<b>17</b>
<b>1.3</b>	<b>Objetivos</b>	<b>17</b>
1.3.1	Objetivos específicos	17
<b>1.4</b>	<b>Estrutura do trabalho</b>	<b>19</b>
<b>2</b>	<b>REFERENCIAIS TEÓRICOS</b>	<b>20</b>
<b>2.1</b>	<b>Indústria 4.0</b>	<b>20</b>
2.1.1	Braço robótico	22
2.1.2	O braço robótico ED-7220C	22
<b>2.2</b>	<b>Computação em Nuvem</b>	<b>24</b>
2.2.1	Nuvem Pública	25
2.2.2	Nuvem Privada	25
2.2.3	Nuvem Híbrida	26
2.2.4	Plataformas de Computação em Nuvem	26
<b>2.3</b>	<b>Trabalhos Relacionados</b>	<b>26</b>
<b>3</b>	<b>METODOLOGIA</b>	<b>28</b>
<b>4</b>	<b>DESENVOLVIMENTO</b>	<b>30</b>
<b>4.1</b>	<b>Visão geral</b>	<b>30</b>
<b>4.2</b>	<b>Conexão local</b>	<b>30</b>
<b>4.3</b>	<b>Nuvem Privada</b>	<b>32</b>
<b>4.4</b>	<b>Nuvem Pública</b>	<b>34</b>
<b>4.5</b>	<b>Nuvem Híbrida</b>	<b>37</b>
<b>5</b>	<b>APRESENTAÇÃO E ANÁLISE DOS RESULTADOS</b>	<b>40</b>
<b>5.1</b>	<b>Conexão local</b>	<b>40</b>
<b>5.2</b>	<b>Nuvem Privada</b>	<b>43</b>
<b>5.3</b>	<b>Nuvem Pública</b>	<b>46</b>
<b>5.4</b>	<b>Nuvem Híbrida</b>	<b>49</b>
<b>5.5</b>	<b>Análise dos Resultados</b>	<b>52</b>
<b>6</b>	<b>CONCLUSÃO</b>	<b>56</b>
<b>6.1</b>	<b>Limitações e trabalhos futuros</b>	<b>56</b>

REFERÊNCIAS . . . . . 58

# 1 Introdução

Atualmente, as empresas buscam cada vez mais ganhar espaço no mercado e uma forma de alcançar um grupo maior de usuários é tornar os serviços o mais virtual possível utilizando sistemas inteligentes. Com o auxílio da Computação em Nuvem, serviços da Indústria 4.0 se fazem úteis no dia a dia industrial.

Um dos pontos principais da Indústria 4.0 é a Computação em Nuvem, que fornece um grande conjunto de fontes para armazenar, processar e analisar dados, gerando informações precisas e reduzindo custos. A Computação em Nuvem é uma tecnologia que oferece suporte à infraestrutura de Tecnologia da Informação (TI) online. Em um ambiente industrial a Computação em Nuvem auxilia na integração de equipamentos inteligentes, favorecendo a intercomunicação dentro e fora do mesmo. A Computação em Nuvem tem expandido a plataforma e a gama de recursos disponíveis para aplicação industrial, permitindo rapidez no processamento de dados, armazenamento e economia de recursos.

As vantagens da adoção da Computação em Nuvem pela Indústria 4.0 são inúmeras, tais como a flexibilidade e o compartilhamento de dados facilitado, já que as informações não estão alocadas fisicamente, várias pessoas e em qualquer local podem acessar essas informações; o tempo de obtenção de informações e respostas que é otimizado através da centralização de dados e virtualização do armazenamento; o monitoramento, pois a Computação em Nuvem fornece controle de operações e rastreamento rápido de funcionalidades permitindo que as empresas tomem decisões de forma rápida; e a economia, já que terceiriza gastos de manutenção e suporte técnico (PAZ; LOOS, 2020).

## 1.1 Problema

Com o emprego da tecnologia no ambiente industrial, os processos se tornam cada vez mais automatizados e computadorizados com a aplicação da Computação em Nuvem. A utilização da Computação em Nuvem na indústria permite ultrapassar limites físicos dos servidores presentes na empresa, permitindo também ampliar as possibilidades de conectividade entre os sistemas utilizados internamente e externamente (VINICIUS et al., 2019).

Para aplicar a Computação em Nuvem no meio industrial, precisamos levar em conta questões como confiabilidade, flexibilidade, usabilidade e segurança. Portanto, para que possamos entender, aplicar e classificar os tipos de nuvem disponíveis no mercado serão analisados como pontos problemáticos a conectividade, tempo de processamento e

resposta na comunicação cliente-servidor. Neste trabalho, a confiabilidade e agilidade das nuvens são colocadas em questão.

## 1.2 Justificativa

A utilização da Computação em Nuvem torna possível a descentralização do sistema utilizado pela empresa, a terceirização e flexibilização de componentes e serviços. A economia gerada pela utilização desses serviços terceirizados, seja economia monetária ou de espaço físico, é real. O fato de se utilizar um produto pronto oferecido por uma empresa especializada é um ponto a ser levado em consideração.

As consequências da utilização dos serviços da Computação em Nuvem podem ser positivas e/ou negativas. No caso positivo, pode-se citar o ganho de tempo, já que o contratante não vai precisar desenvolver um serviço do começo, buscando exatamente a solução necessária ao seu negócio. A economia de espaço físico, onde o contratante deixa de comprar equipamentos, as vezes de grande porte, utilizando o serviço de armazenamento e execução de serviços computacionais da empresa contratada é uma outra grande vantagem. Já como casos negativos, podemos citar o fato de que esses serviços oferecidos podem custar caro, bem como a terceirização da segurança dos dados e execuções importantes da própria empresa.

## 1.3 Objetivos

O objetivo geral deste trabalho é instalar, configurar e simular serviços de Computação em Nuvem selecionados e realizar testes comparativos de conexão entre os mesmos, levantando dados característicos acerca dos tipos de nuvens utilizadas e por fim, realizar a avaliação e comparação entre eles.

### 1.3.1 Objetivos específicos

Tendo como base um ambiente similar à uma Indústria 4.0, serão implementadas plataformas de Computação em Nuvem que estão disponíveis no mercado e que são aplicáveis à situação, e então, serão realizados testes para cada um dos pontos críticos de rede levantados utilizando ferramentas específicas. Para cada uma das plataformas, serão realizados testes da seguinte forma:

- Conectividade: serão analisadas as conexões entre dispositivos e os servidores a partir da rede disponível;
- Processamento: medido como o tempo que determinado dado leva para ser processado nos servidores utilizados;

- Latência: atraso observado entre o contato e a resposta entre os servidores e usuários bem como dos *datacenters* mais próximos.



## 1.4 Estrutura do trabalho

Este trabalho está organizado de acordo os seguintes capítulos: o capítulo 2 descreve os mecanismos e serviços utilizados (sistemas operacionais, *softwares* de apoio e aplicações das nuvens selecionadas) para desenvolver o trabalho bem como os trabalhos relacionados ao mesmo. No capítulo 3, é descrito como foi realizado o trabalho e o levantamento de informações e ferramentas utilizadas. O capítulo 4 descreve o passo a passo do desenvolvimento do trabalho, instalação das plataformas, ferramentas de teste e como os mesmos foram realizados. O capítulo 5 mostra os experimentos realizados bem como os resultados obtidos. Por fim, o capítulo 6 mostra as conclusões obtidas pelos testes e o comparativo entre os serviços testados.

## 2 Referenciais Teóricos

A Indústria 4.0, dita como sendo a quarta grande evolução industrial, utiliza de sistemas inteligentes com alto grau de automação e tomada de decisões autônomas, que com o avanço da TI e dos objetos inteligentes, tornam processos industriais mais rápidos e precisos.

A Computação em Nuvem é um modelo de computação que permite aos usuários acessarem aplicações e serviços a qualquer hora, em qualquer lugar e independente da plataforma, desde que estejam conectado à rede da nuvem. Dispositivos inteligentes precisam acessar os recursos que devem estar disponíveis através da rede. Os dados que uma indústria gera precisam ser processados, armazenados e atualizados com frequência. Cada vez mais o processamento e o armazenamento estão sendo movidos dos computadores pessoais para grandes provedores de serviços nas nuvens. A velocidade de processamento é um quesito muito importante para o desenvolvimento financeiro de uma empresa. O processamento leva em conta a capacidade do sistema de entregar resultados em tempo hábil, dado que a quantidade de dados de entrada pode ser grande. Quanto mais dispositivos conectados à rede maior a probabilidade de ocorrência de congestionamento e aumento da latência ([PEDROSA; NOGUEIRA, 2011](#)).

Para uma eficiente gravação de dados, uma indústria deve dispor de bancos de dados confiáveis. Esses bancos de dados devem oferecer armazenamento de dados estruturados e processamento rápido e confiável de transações ([POSSOBOM, 2012](#)). O armazenamento diretamente nos servidores evita atrasos causados pela distância e os meios de comunicação utilizados, assim, o tempo que os clientes levam para realizar a troca de informações com o servidor e equipamentos não devem permitir espera no processamento.

As plataformas de serviço em nuvem contêm muitos mecanismos de processamento e segurança com alta confiabilidade para fornecer um serviço seguro. A partir do fato de que uma empresa trabalha com dados importantes e confidenciais, ela precisa possuir um esquema de segurança apropriado a fim de que todos esses dados sejam mantidos seguros. A nuvem é uma fonte muito grande de informações importantes que nenhuma empresa pode controlar completamente, vazamento de dados e ataques aos sistemas se tornaram uma grande ameaça para muitos domínios ([YUE et al., 2015](#)).

### 2.1 Indústria 4.0

O século XVIII ficou marcado no mundo da manufatura pelo início do emprego das máquinas a vapor, sendo assim caracterizada a Indústria 1.0. Logo depois, com o

surgimento e o emprego da eletricidade na indústria foi denominada como Indústria 2.0. Com o surgimento da TI, possibilitando o emprego da telecomunicação e robótica na indústria, tivemos o surgimento da Indústria 3.0 (INOUE et al., 2019).

O termo “Indústria 4.0”, ficou conhecido publicamente em 2011 na Alemanha, a partir de uma iniciativa chamada “Industrie 4.0” - uma associação de representantes de empresas, políticos e acadêmicos - que apoiava a ideia como uma abordagem para fortalecer a competitividade entre indústrias alemãs (HERMANN; PENTEK; OTTO, 2016). Na Indústria 4.0 são utilizadas várias tecnologias de ponta em diversas áreas, como Computação em Nuvem e robôs automatizados. A Indústria 4.0 vai muito além da automação do processo industrial, ela envolve a integração e comunicação entre tecnologias físicas e digitais que fazem parte do sistema.

O grande salto dado pelo surgimento da Indústria 4.0 foi o fato de que os processos evoluíram de apenas uma célula automatizada para sistemas automatizados e integrados que se comunicam entre si e com recursos externos de qualquer lugar do mundo, fornecendo assim flexibilidade, velocidade de comunicação entre as pontas, produtividade e qualidade dos sistemas produtivos proporcionados pelos grandes avanços tecnológicos da área de TI e engenharias (ALBERTIN et al., 2017).

Como importantes braços da Indústria 4.0 devemos citar (ALBERTIN et al., 2017):

- IoT: integração entre objetos físicos e virtuais conectados à internet permitindo a coleta, troca de informações e armazenamento de dados, que após processamento, geram informações capazes de otimizar o relacionamento entre humanos e máquinas;
- Realidade Aumentada (RA): definida como a variação de ambientes virtuais, permitindo a sobreposição de objetos reais e virtuais compondo um ambiente em comum;
- Robôs Autônomos: robôs inteligentes e cooperativos, capazes de operar com autonomia no mundo real sem qualquer controle externo explícito;
- Simulação: imitação da operação original de um processo a partir da observação do mesmo, gerando assim resultados fieis aos naturais, podendo assim extrair suposições sobre as características operacionais do sistema real que está sendo simulado;
- Manufatura Aditiva: técnica automatizada para conversão de dados CAD 3D em objetos físicos (impressão 3D);
- *Big Data*: grande quantidade de dados e informações que são gerados no processo de produção pelos equipamentos de fabricação inteligente;
- *CyberSecurity*: segurança virtual dos dados e serviços contra ataques e ameaças;
- Nuvem: sistema de computação que provê recursos e serviços baseados em servidores remotos.

### 2.1.1 Braço robótico

Com a evolução e inclusão da tecnologia na indústria, tem se utilizado cada vez mais máquinas automáticas e braços robóticos no processo industrial. Um braço robótico é um equipamento programável que realiza funções semelhantes às de um braço humano, porém, repetidas vezes e com maior precisão. O movimento do braço robótico é determinado por motores que acionam e movimentam os seus componentes, realizando assim, diferentes ações programadas.

### 2.1.2 O braço robótico ED-7220C

O braço robótico utilizado neste trabalho é o ED-7220C, exibido na figura 1, desenvolvido pela empresa coreana *ED-Laboratory* com fins exclusivamente didáticos e possui como controlador padrão o ED-MK4. Como o mesmo não dispunha de conexão com a internet e nem da possibilidade de adição de novos componentes de *hardware*, (RODRIGUES, 2022) em seu projeto desenvolveu uma plataforma de controle mais moderna para o equipamento.



Figura 1 – Braço robótico ED-7220C  
Fonte: (RODRIGUES, 2022)

As características básicas do braço robótico ED-7220C utilizado estão descritas na Tabela 1.

Componente	Descrição
Número de juntas	5 Juntas + Garra
Construção	Braço Articulado Vertical
Quantidade de Switch's	5
Precisão (Posição)	$\pm 0.5\text{mm}$
Velocidade (movimento)	100mm/s max
Carga Máxima	1Kg
Atuador	Motor DC com encoder óptico
Range de Movimento	<ul style="list-style-type: none"> <li>• Base: <math>310^\circ</math></li> <li>• Ombro: <math>+130^\circ / -35^\circ</math></li> <li>• Cotovelo: <math>\pm 130^\circ</math></li> <li>• Garra (rotação): <math>360^\circ</math></li> </ul>
Abertura da Garra	55 mm
Dimensão	<ul style="list-style-type: none"> <li>• Base: <math>220^\circ \times 180(\text{H})\text{mm}</math></li> <li>• Braço: <math>220+220\text{mm}</math></li> </ul>
Peso	33Kg

Tabela 1 – ED-7220C: Características  
 Fonte: (RODRIGUES, 2022)

A composição física de um manipulador robótico articulado é dada por elos e *links*, tentando simular de forma robótica os movimentos de um braço humano. Através do movimento realizado pelo braço robótico, se classifica o mesmo pelos graus de liberdade, sendo esse definido pela quantidade de parâmetros independentes que compõem o braço (RODRIGUES, 2022). O braço robótico usado, possui seis graus de liberdade, assim como um braço humano. A figura 2 mostra a classificação e os movimentos do braço robótico ED-7220C.

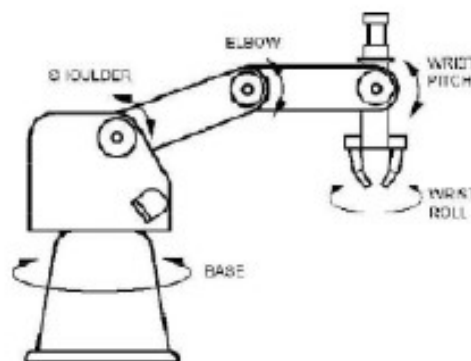


Figura 2 – Graus de liberdade do braço robótico  
 Fonte: (RODRIGUES, 2022)

O movimento do braço é determinado através de motores que são responsáveis pelo movimento de cada componente do mesmo, fazendo com que ele atinja uma determinada

posição no espaço e realize a ação desejada. Cada motor possui um *encoder* óptico cujo sinal é utilizado como entrada na unidade controladora (RODRIGUES, 2022).

O movimento do braço é descrito pela cinemática, utilizada para modelar as equações de transformação entre os ângulos das juntas do robô e o acionador final no espaço cartesiano. Dividida em duas categorias, a cinemática é caracterizada por Cinemática Direta (*Forward Kinematic* (FK)) e Cinemática Inversa (*Inverse Kinematic* (IK)) (RODRIGUES, 2022). A Cinemática Direta é usada para obter a localização do acionador no plano cartesiano com base nos ângulos de cada junta do robô, porém é pouco útil. Já a Cinemática Inversa informa qual o ângulo relativo de cada *link* para que o acionador alcance um determinado ponto no espaço cartesiano.

## 2.2 Computação em Nuvem

A Computação em Nuvem é um modelo de rede que permite o acesso a um grande conjunto de recursos de computação que são compartilhados, configurados e podem ser solicitados e fornecidos rapidamente com o menor esforço possível por parte do usuário, demandando mínima interação com o provedor de serviços (PAZ; LOOS, 2020). Como citado por (PAZ; LOOS, 2020), podemos classificar esse modelo de Computação em Nuvem em cinco características essenciais, três modelos de implantação e três modelos de serviço:

Características essenciais:

- Serviço *on-demand*: onde o usuário tem acesso aos serviços onde e quando necessário;
- Amplo acesso à rede: os serviços são acessíveis de qualquer plataforma;
- Conjunto agrupado de recursos: mesmo que alocados em diferentes *datacenters*, os recursos são agrupados e oferecidos de forma eficiente;
- Rápida elasticidade: facilidade e liberdade de alocação de maior e menor conjunto de serviços e recursos;
- Serviço mensurável: controle e monitoramento dos serviços pela nuvem.

Modelos de implantação:

- Nuvem privada: acessível dentro de um ambiente protegido por *firewall*;
- Nuvem pública: acessível em toda a internet e à vários usuários simultaneamente, mas separadamente;
- Nuvem híbrida: composta por duas ou mais nuvens que preservem características originais de seu modelo.

Modelos de serviço:

- Infraestrutura como serviço (IaaS): disponibilização de máquinas virtuais com recursos de processamento, armazenamento, servidores e componentes de rede;
- Plataforma como serviço (PaaS): disponibilização de sistema operacional, linguagens de programação e ambiente de desenvolvimento de aplicações;
- *Software* como Serviço (SaaS): disponibilização de *software*, assim o usuário não precisa adquirir licença do mesmo, pagando apenas o que utiliza pelo tempo necessário.

A Computação em Nuvem é uma ferramenta dinâmica em que seus recursos são provisionados à medida que a demanda cresce ou diminui dentro de um modelo que é baseado em métricas de uso (INOUE et al., 2019). Os custos de adoção e uso da Computação em Nuvem são calculados, na maioria das vezes, de acordo com seu uso ou serviço adquirido, tudo proporcionalmente ofertado.

### 2.2.1 Nuvem Pública

Dizemos que a nuvem é do tipo pública quando seus recursos como *hardware* e *software* são compartilhados entre diversos usuários simultaneamente, porém, cada serviço de cada cliente é hospedado de forma individual, mantendo sempre seus processos privados e acessíveis por seus usuários.

Nesta configuração de nuvem, apenas a infraestrutura é compartilhada. A nuvem é implementada pelo prestador de serviços, que assume a responsabilidade e garantia de desempenho e de segurança da mesma (PEDROSA; NOGUEIRA, 2011). Como sua característica lhe permite compartilhamento de hardware entre os usuários, seu custo é relativamente baixo, porém, característica essa que lhe dá também o título de pouco segura, o que demanda a utilização de recursos e aplicações extras para lhe garantir maior segurança.

### 2.2.2 Nuvem Privada

A nuvem privada leva esse nome justamente por ser limitada a um determinado grupo de usuários, normalmente usada dentro de organizações, sendo operadas exclusivamente para estas (PEDROSA; NOGUEIRA, 2011). Os recursos disponibilizados são exclusivos para cada usuário, podendo inclusive serem hospedados em suas própria infraestrutura e gerenciados pela própria organização ou por terceiros (PEDROSA; NOGUEIRA, 2011).

A maior característica desse tipo de nuvem é a segurança, possuindo estrutura mais completa e dispor de maior diversidade de configurações, seu uso e manutenção são de maior custo.

### 2.2.3 Nuvem Híbrida

O serviço de nuvem híbrida é descrito como a junção dos serviços públicos e privados de Computação em Nuvem. Essa combinação de serviços proporciona benefícios e versatilidade de recursos, podendo utilizar de cada um para determinada aplicação ou serviço. Quando combinadas nessa modalidade, cada nuvem se mantém como uma entidade individual, mantendo suas características, porém, estão unidas pelo uso de tecnologia proprietária ou padronizada, garantindo a portabilidade de dados e aplicações (PEDROSA; NOGUEIRA, 2011). Vale ressaltar que a nuvem híbrida não garante todas as funcionalidades de nenhuma das nuvem que a compõe.

### 2.2.4 Plataformas de Computação em Nuvem

Foram levantadas e estudadas um conjunto de plataformas de Computação em Nuvem e destas foram selecionadas, devido a maior simplicidade de instalação e manipulação, bem como custo e disponibilidade, as seguintes para aplicação no trabalho:

- *OpenNebula*: plataforma de computação em nuvem que possibilita a gerência de infraestruturas heterogêneas virtuais de um *datacenter* a fim de criar uma implementação privada de nosso serviço. O *OpenNebula* oferece uma solução simples e flexível para construção e gerenciamento de nuvens empresariais e sua estrutura básica se dá pelo sistema *frontend*, rede de serviços, *datastores* e *hosts*.
- *Google Cloud Platform*: serviço de Computação em Nuvem que oferece serviços e ferramentas que auxiliam no desenvolvimento, implementação, entrega e gerenciamento de aplicações. Dentre os serviços oferecidos, destacam-se: armazenamento, serviços de rede, gestão e desenvolvimento de aplicações.

## 2.3 Trabalhos Relacionados

No trabalho realizado por (SEGALIN, 2013) são realizados testes que envolvem o processamento de serviços em nuvem e como eles são afetados com a elasticidade, ou seja, a alteração da configuração de recursos da máquina virtual utilizada.

(SILVA et al., 2020) em seu trabalho, realiza testes de latência em um serviço de *Cloud Gaming*, em situações semelhantes às do presente trabalho, como conexão local e nuvem, resultando em características semelhantes para as mesmas: menor latência para conexão local e maior para serviços na nuvem, comprovadamente.



No trabalho *Cloud Test: melhoria da confiabilidade de testes mobile com a utilização de computação em nuvem*, ([OLIVEIRA, 2021](#)) demonstra testes *mobile* através da Computação em Nuvem e como os resultados foram otimizados utilizando tais plataformas.

Diferentemente dos trabalhos relacionados apresentados, o presente trabalho testará os serviços de Computação em Nuvem selecionados em um ambiente de uma Indústria 4.0 e irá classificá-los a partir dos resultados obtidos.

### 3 Metodologia

Para simular um braço robótico modernizado, foram utilizados códigos desenvolvidos por (RODRIGUES, 2022) em seu trabalho, utilizando a linguagem Python. Como os códigos tinham a finalidade de serem utilizados em um braço robótico, os mesmos precisaram ser adaptados para que pudessem ser executados sem a necessidade do posicionamento original do braço, sendo enviadas somente as coordenadas simuladas.

No funcionamento real do braço robótico, foi utilizada uma câmera para tirar uma fotografia do espaço de trabalho do manipulador robótico e então, através de um algoritmo de detecção de objetos são identificadas as posições de cada objeto neste espaço (RODRIGUES, 2022). No presente trabalho, é utilizada uma fotografia deste espaço de trabalho com os objetos dispostos, mostrada na Figura 3. Essa imagem é enviada pelo algoritmo cliente que é executado na máquina local ao algoritmo servidor que é executado na nuvem, onde então é processado pelo algoritmo de detecção de objetos e identificadas as suas respectivas posições. Importante ressaltar que essa adaptação não impacta nos resultados.

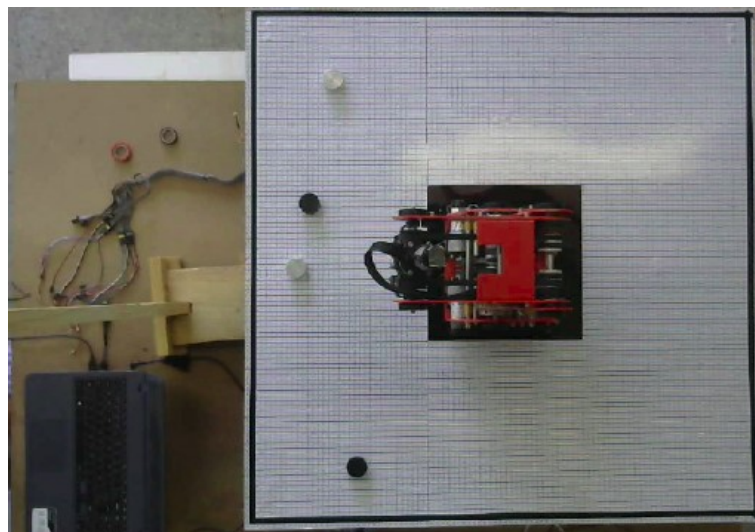


Figura 3 – Fotografia do espaço de trabalho  
Fonte: (RODRIGUES, 2022)

Foi escolhido o protocolo *User Datagram Protocol* (UDP) para realizar a conexão cliente-servidor, apesar de não ser um protocolo confiável, é simples e proporciona uma boa performance de processamento pela rapidez. O protocolo *Transmission Control Protocol* (TCP) poderia ser utilizado para realizar a conexão, porém, dada sua complexidade foi descartada a possibilidade, já que o protocolo UDP realiza a tarefa necessária oferecendo maior simplicidade para esta aplicação e que a própria nuvem pode tratar a questão de confiabilidade.

A simulação se deu pela utilização da nuvem como sendo servidor e a máquina local como cliente executando o código do braço robótico, cada um executando o algoritmo referido para estabelecer a comunicação cliente-servidor. O servidor é executado e permanece escutando pelo cliente, até que o mesmo estabeleça a conexão. O cliente envia ao servidor uma imagem que simula a visão da câmera no espaço de trabalho. Essa imagem, mostrada na Figura 3, possui quatro peças, porém, o algoritmo só reconhece as de cor preta e prata, com isso, possui três peças válidas dispostas no espaço, cada uma com suas referidas coordenadas. O servidor recebe essa imagem e calcula por meio desta e de algoritmos de cálculo de coordenadas, a posição relativa de cada uma das peças e envia para o cliente o resultado.

Com cada conexão estabelecida, são feitos três cálculos relacionados a cada uma das três peças dispostas no espaço de trabalho, o algoritmo do servidor calcula o tempo de processamento de cada uma das três peças e o algoritmo do cliente calcula a latência de cada uma das três conexões e o tempo total de execução do processo.

Para cada serviço de Computação em Nuvem selecionado, foram executados 10 vezes os testes para gerar uma média das variações e então avaliados os resultados.

## 4 Desenvolvimento

### 4.1 Visão geral

Para o desenvolvimento do trabalho, foi utilizado um notebook *Dell Vostro 15*, modelo P75F com 8GB de ram, SSD de 256Gb e processador *Intel(R) core(TM) i5-8265U CPU @ 1.60GHz 1.80GHz* utilizando como sistema operacional *Windows 11 Home Single Language* Versão 22H2 para a execução do papel cliente na conexão cliente-servidor, trabalhando como se fosse o controlador do braço robótico.

O algoritmo de controle do braço robótico foi desenvolvido em Python e foi criado para atender aos requisitos de funcionamento do ED-7220C. A linguagem Python foi escolhida por ser uma linguagem moderna, sendo ideal para automatizar processos que são repetitivos e também amplamente empregada em aplicações *web*, administração de sistemas e interfaces gráficas, bem como em ciência de dados (RODRIGUES, 2022).

Após instalar e configurar a IDE *PyCharm 2022.3.2 (Community Edition)*, o Python 2.7 e as bibliotecas necessárias para a execução dos códigos, foram feitas diversas adaptações nos mesmos para que pudessem ser executados sem problemas e nos retornar variáveis importantes para o trabalho, como a impressão do tempo de execução e latência do processo. Através de uma imagem do espaço de trabalho, Figura 3, que é enviada para o algoritmo de detecção de objetos, são identificadas as posições de cada um dos objetos na imagem. O algoritmo de controle do braço robótico recebe os pontos nos quais os objetos estão localizados no espaço e executa os devidos cálculos para determinar o posicionamento. O código foi setado para um loop de 10 execuções, sendo realizadas então 30 conexões ao servidor local (cada execução processa três objetos na imagem).

### 4.2 Conexão local

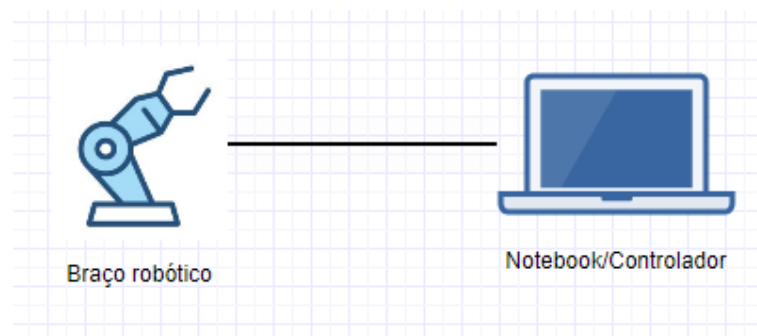
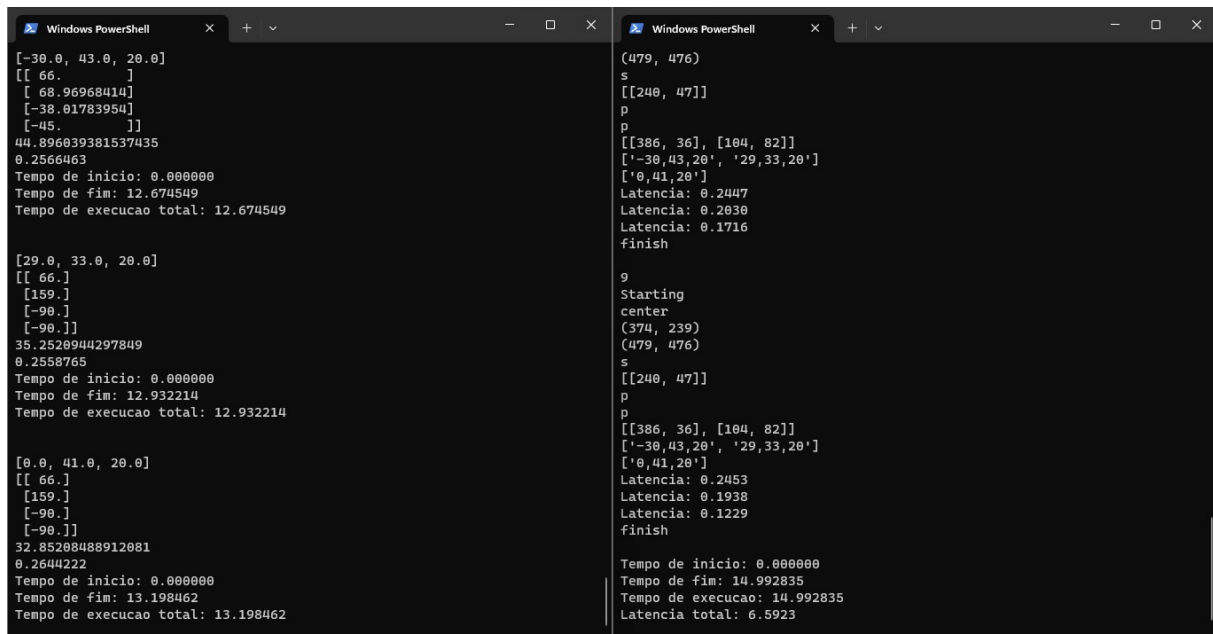


Figura 4 – Diagrama de rede local

Fonte: elaborado pelo autor

Antes da realização do experimento na nuvem, foram executados testes locais usando *localhost* para a conexão cliente-servidor. Como ilustrado no diagrama da Figura 4, o teste simula a conexão direta entre a máquina local e o braço robótico.

A Figura 5 nos mostra como exemplo o retorno da execução do código via *Prompt de Comando* do *Windows*, sendo a janela à esquerda correspondente ao código servidor e à direita correspondente ao código cliente. O cliente envia ao servidor as coordenadas de cada objeto, que faz os cálculos de posicionamento e os imprime na tela. São impressos também o tempo de início (tempo 0.00) e o de término de cada processamento considerando o tempo total da execução para cada objeto. Ao retornar os valores para o cliente, o mesmo imprime o novo posicionamento do braço e a latência para cada objeto calculado. Ambos algoritmos retornam o tempo total de execução e somente o código cliente exibe latência total do processo.



```
Windows PowerShell
[[-30.0, 43.0, 20.0]
[[ 66. ]
[ 68.96968414]
[-38.01783954]
[-45. ]]
44.896039381537435
0.2566463
Tempo de inicio: 0.000000
Tempo de fim: 12.674549
Tempo de execucao total: 12.674549

[29.0, 33.0, 20.0]
[[ 66. ]
[159. ]
[-90. ]
[-90. ]]
35.2520944297849
0.2558765
Tempo de inicio: 0.000000
Tempo de fim: 12.932214
Tempo de execucao total: 12.932214

[0.0, 41.0, 20.0]
[[ 66. ]
[159. ]
[-90. ]
[-90. ]]
32.85208488912081
0.2644222
Tempo de inicio: 0.000000
Tempo de fim: 13.198462
Tempo de execucao total: 13.198462

Windows PowerShell
(479, 476)
s
[[240, 47]]
p
p
[[386, 36], [104, 82]]
['-30,43,20', '29,33,20']
['0,41,20']
Latencia: 0.2447
Latencia: 0.2030
Latencia: 0.1716
finish

9
Starting
center
(374, 239)
(479, 476)
s
[[240, 47]]
p
p
[[386, 36], [104, 82]]
['-30,43,20', '29,33,20']
['0,41,20']
Latencia: 0.2453
Latencia: 0.1938
Latencia: 0.1229
finish

Tempo de inicio: 0.000000
Tempo de fim: 14.992835
Tempo de execucao: 14.992835
Latencia total: 6.5923
```

Figura 5 – Execução na conexão local

Fonte: elaborado pelo autor

Após validado o teste na conexão local, foram então iniciados os testes na nuvem, onde hospedamos o servidor e na máquina local executamos o cliente.

### 4.3 Nuvem Privada

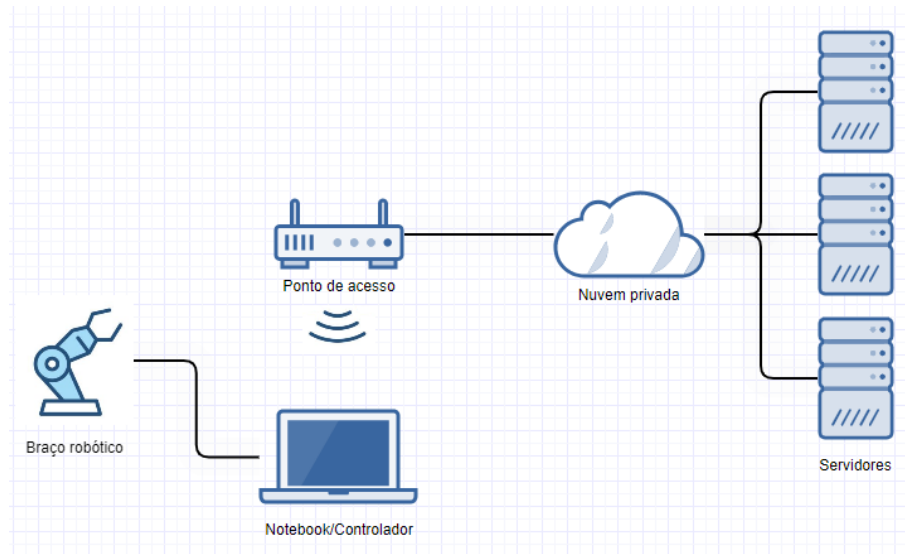


Figura 6 – Diagrama da nuvem privada

Fonte: elaborado pelo autor

O diagrama da Figura 6 ilustra a configuração simulada para a nuvem privada: a máquina local conectada ao braço robótico, porém, o processamento das informações necessárias à movimentação do braço são realizadas na nuvem.

Para a instalação da nuvem *OpenNebula*, foi utilizado o Debian, sistema operacional baseado em Linux, versão 9 (*Stretch*) disponível em (DEBIAN, 2022). Foi utilizado o tutorial disponível no site oficial do *OpenNebula* para a instalação da versão 5.6.1-1, (SYSTEMS, 2022). Para criar e gerenciar a máquina virtual, foi utilizado o *software Virtual Machine Manager* versão 1.4.0 disponível em (CONTRIBUTOR, 2022).

Com todos os *softwares* necessários corretamente instalados e configurados, foi então executada a máquina virtual, através do QEMU/KVM, para disponibilizar o *frontend*. Com a máquina virtual em funcionamento, foram executados no terminal os códigos de ativação do OpenNebula, para que o mesmo possa ser acessado através do navegador no *backend*, como mostrado na Figura 7.

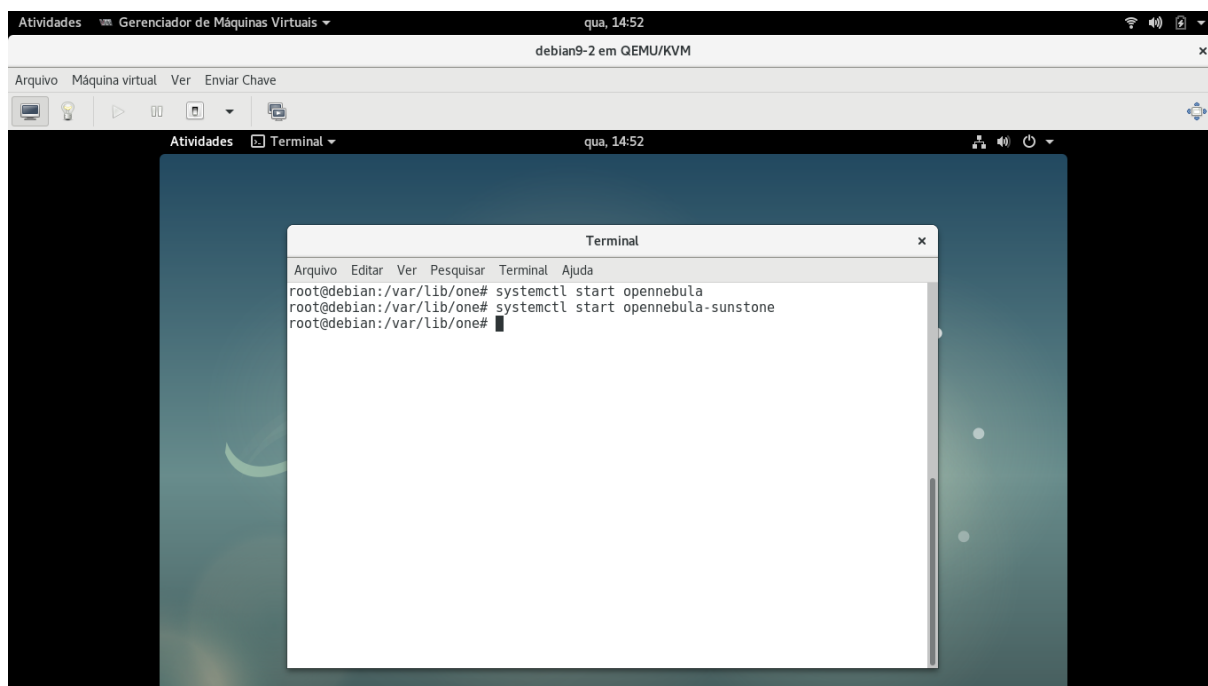


Figura 7 – Ativação do OpenNebula

Fonte: elaborado pelo autor

Através do endereço de IP da máquina virtual e a porta configurada, acessamos a interface do *OpenNebula* utilizando os dados de *login* cadastrados, tendo assim, acesso às informações de máquinas virtuais geradas através do *OpenNebula*, como mostrado nas Figuras 8 e 9 respectivamente.



Figura 8 – Acesso ao OpenNebula

Fonte: elaborado pelo autor

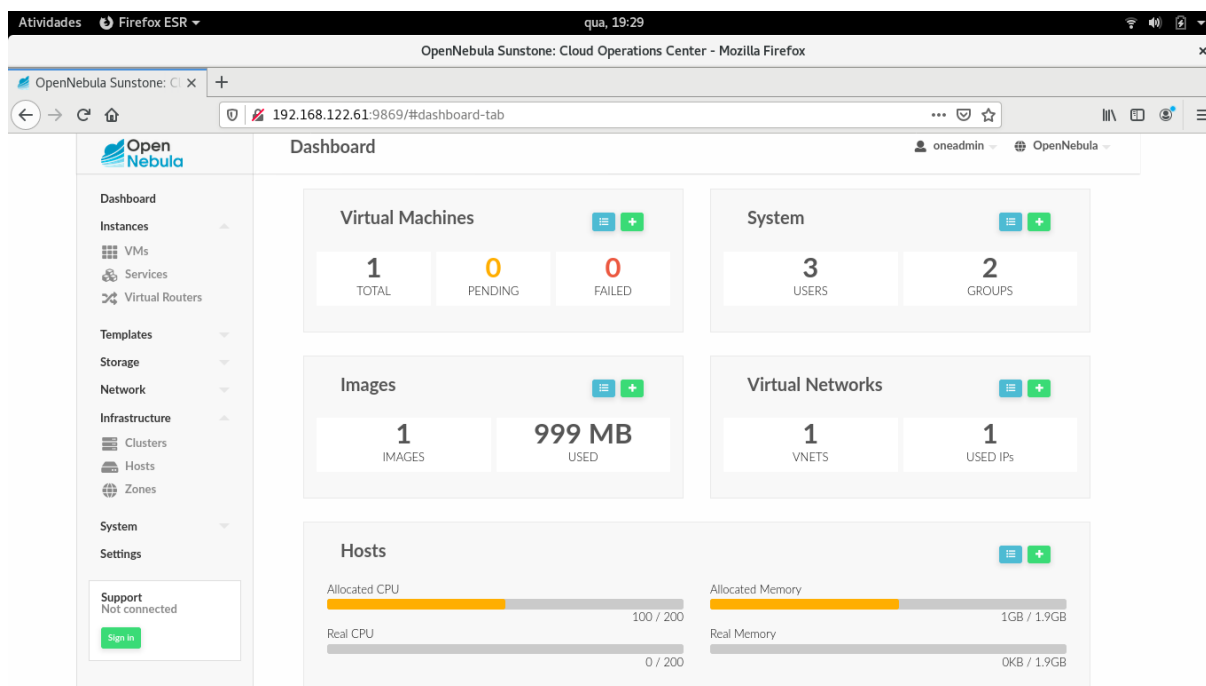


Figura 9 – Interface do OpenNebula

Fonte: elaborado pelo autor

Foi executado o código na máquina virtual instanciada no *OpenNebula* e o código cliente na máquina física para a execução de cada um dos testes. Para obter uma média aceitável dos resultados, os códigos foram executados 10 vezes.

## 4.4 Nuvem Pública

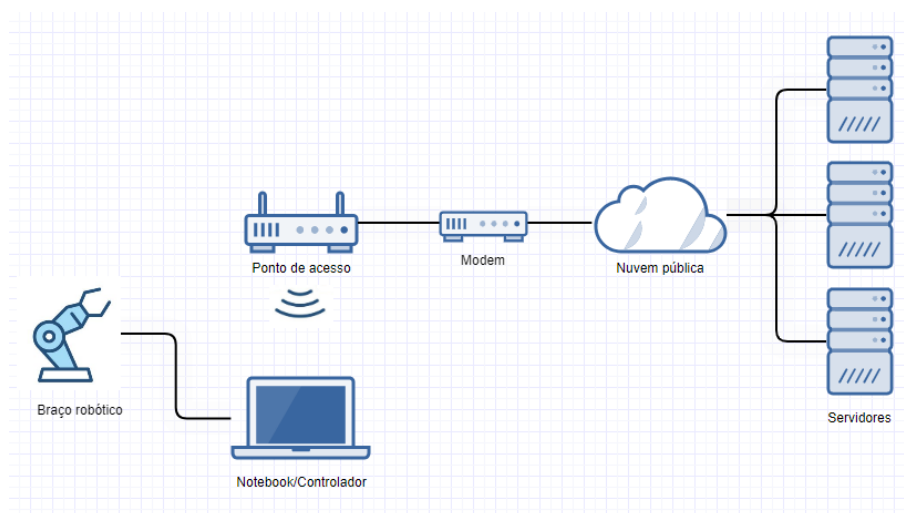


Figura 10 – Diagrama da nuvem pública

Fonte: elaborado pelo autor

A Figura 10 mostra um diagrama onde é ilustrada a configuração simulada para a nuvem pública, onde a máquina local conectada ao braço robótico, envia à nuvem a



solicitação de processamento dos dados e então recebe as informações atualizadas.

Para a utilização da *Google Cloud*, foi utilizado o período de testes disponível em (LLC, 2023), em que são oferecidos créditos para montar o serviço com as configurações desejados em um período de 90 dias. Foi então configurada uma máquina virtual executando a versão 5.15.65+ do Linux x86\_64.

Acessando a página do *Google Cloud* e feitas todas as configurações da nuvem, com a máquina virtual ligada e configurada, como mostra a Figura 11, todas as bibliotecas Python necessárias para a execução do algoritmo foram instaladas via terminal, foi então executado na nuvem o código servidor e na máquina física o código cliente usando o prompt de comando, como mostram as Figuras 12 e 13.

Como feito para a conexão local, foram executados 10 vezes, para obter a média dos valores calculados.

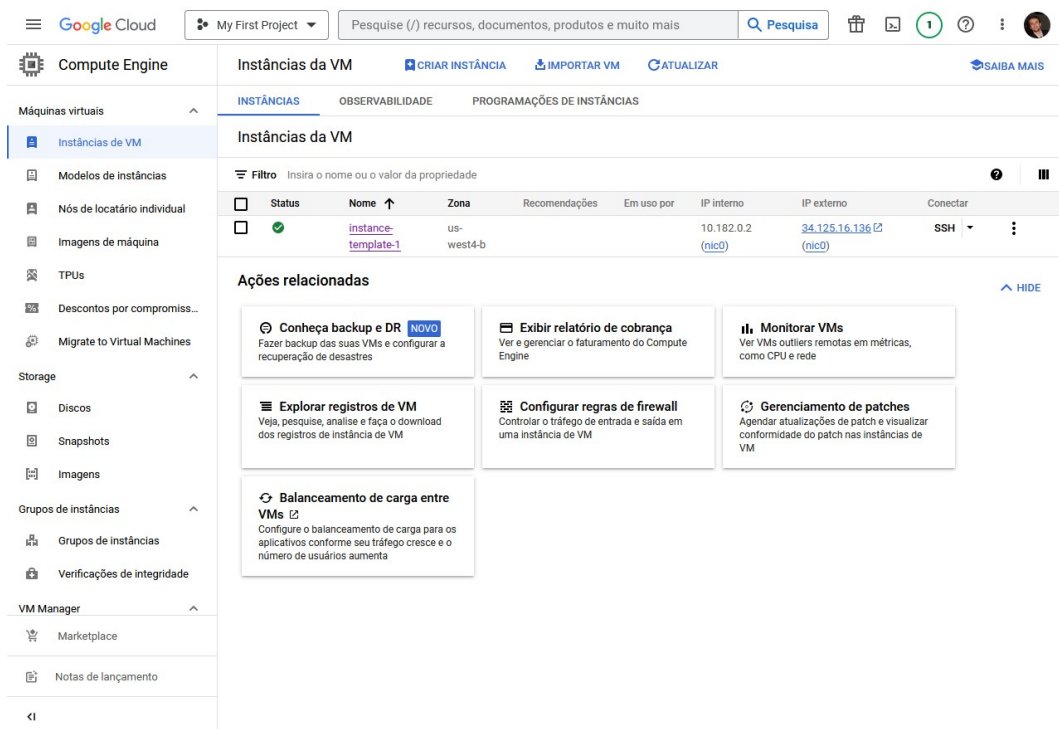


Figura 11 – Interface de Instâncias da VM *Google Cloud*

Fonte: elaborado pelo autor

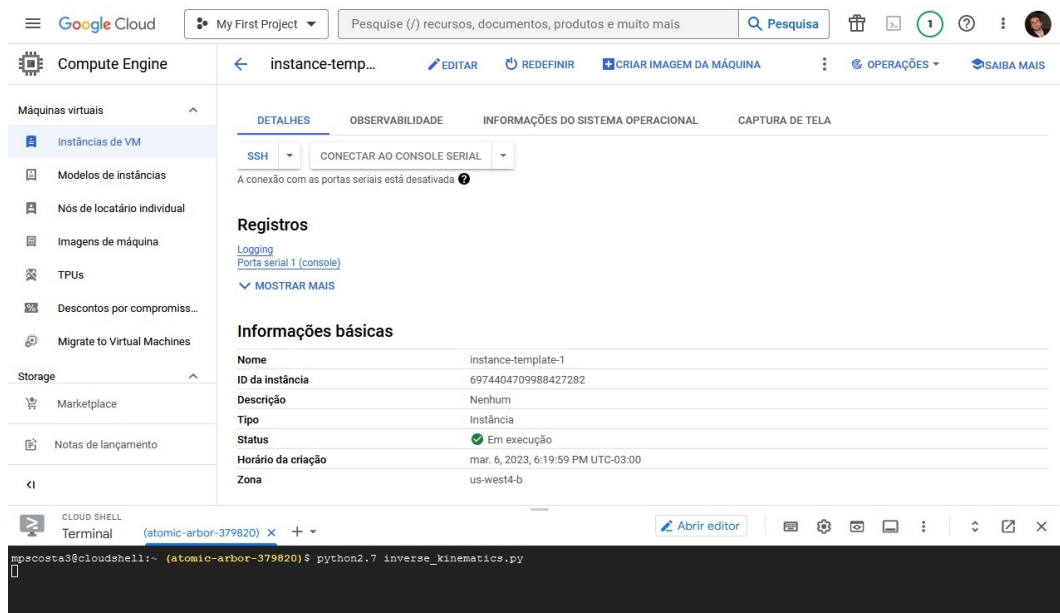


Figura 12 – Servidor aguardando conexão do cliente  
Fonte: elaborado pelo autor

```

Prompt de Comando
[ '-30,43,20', '29,33,20' ]
['0,41,20']
Latencia: 184.2787
Latencia: 178.4202
Latencia: 178.6862
finish

9
Starting
center
(374, 239)
(479, 476)
s
[[240, 47]]
p
p
[[386, 36], [104, 82]]
[ '-30,43,20', '29,33,20' ]
['0,41,20']
Latencia: 179.4217
Latencia: 178.2175
Latencia: 178.0139
finish

Tempo de inicio: 0.000000
Tempo de fim: 20.984624
Tempo de execucao: 20.984624
Latencia total: 5371.7474

C:\Users\marco\Desktop\TCC20223\SEnew\camera>

```

Figura 13 – Cliente conectado à Google Cloud  
Fonte: elaborado pelo autor

## 4.5 Nuvem Híbrida

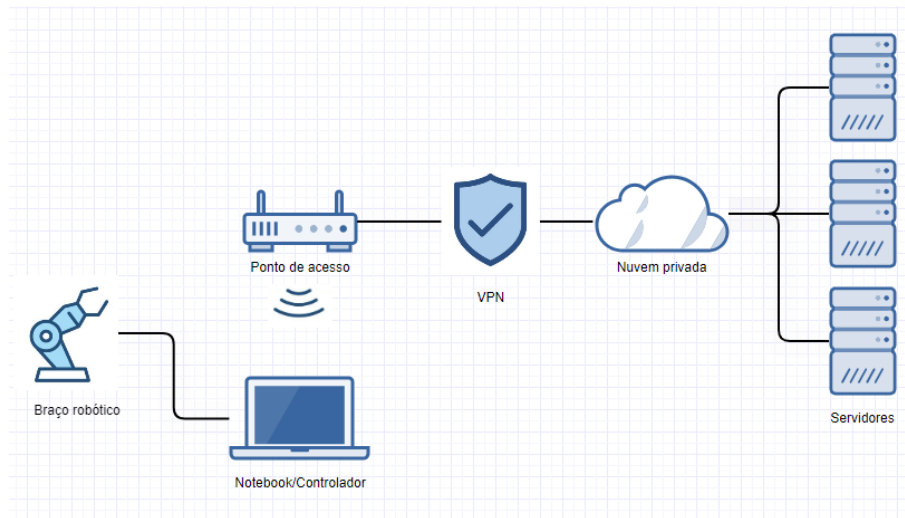


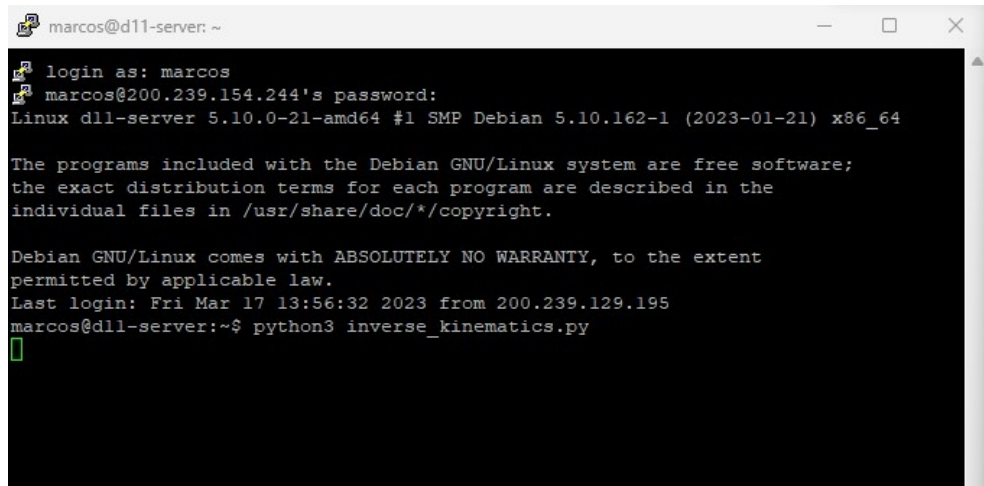
Figura 14 – Diagrama da nuvem híbrida

Fonte: elaborado pelo autor

O diagrama ilustrado na Figura 14 mostra a configuração simulada para a nuvem híbrida: a máquina local conectada ao braço robótico envia à nuvem a solicitação de processamento dos dados e recebe a resposta calculada, tudo através da conexão com a VPN.

Para o teste em nuvem híbrida, foi utilizada a mesma configuração do *OpenNebula* usada no teste de nuvem privada, com o código servidor já alocado na mesma, foi utilizada a rede privada virtual (VPN) da UFOP estabelecendo conexão com a rede da nuvem, acessando assim seus recursos. Mesmo que o braço robótico esteja localizado na UFOP e a VPN utilizada no teste é a da UFOP, são redes diferentes, o acesso à VPN é apenas para realizar a conexão com a nuvem onde é executado o algoritmo de cálculo de posicionamento do braço.

Uma vez conectado à VPN, foi utilizado o *software* Putty, disponível em (VOLUNTEERS, 2023), para fazer a conexão *Secure Shell* (SSH) com o *OpenNebula* e assim, executando o código servidor no mesmo como mostra a Figura 15, e na máquina local, executado o código cliente.

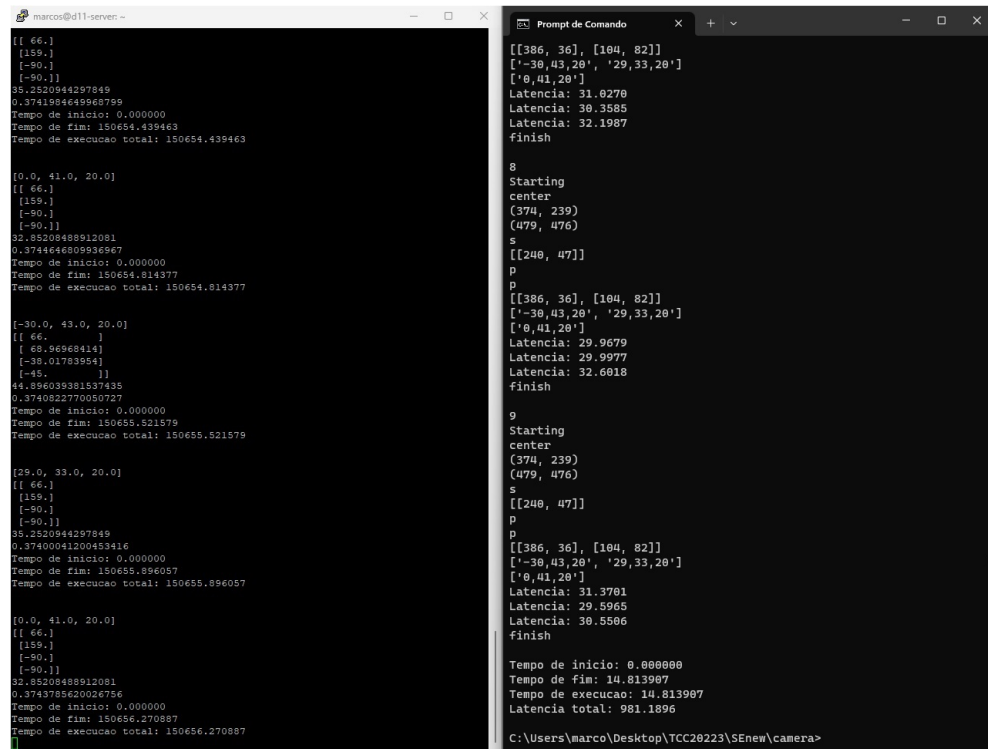


```
marcos@d11-server: ~  
login as: marcos  
marcos@200.239.154.244's password:  
Linux d11-server 5.10.0-21-amd64 #1 SMP Debian 5.10.162-1 (2023-01-21) x86_64  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Fri Mar 17 13:56:32 2023 from 200.239.129.195  
marcos@d11-server:~$ python3 inverse_kinematics.py  
█
```

Figura 15 – Servidor conectado através do Putty

Fonte: elaborado pelo autor

A Figura 16 nos mostra a interação entre os códigos cliente e servidor rodando na máquina local (à direita) e na nuvem híbrida (à esquerda) respectivamente. O cliente envia para o servidor as coordenadas de cada objeto da imagem, que faz os cálculos de seus posicionamento e então os imprime na tela acompanhados pelo tempo de início do processamento (tempo 0.00) e o de término para o processamento de cada objeto considerando o tempo total da execução para cada um deles e retorna para o cliente, que imprime o novo posicionamento e a latência para cada objeto calculado. Ambos algoritmos retornam o tempo total de execução e somente o código cliente exibe a latência total do processo.



```

marcos@d11-server ~
[[ 66.]]
[[ 159.]]
[[ -50.]]
[[ -90.]]
35.2520944297849
0.3741904649960799
Tempo de inicio: 0.000000
Tempo de fim: 150654.439463
Tempo de execucao total: 150654.439463

[0.0, 41.0, 20.0]
[[ 66.]]
[[ 159.]]
[[ -50.]]
[[ -90.]]
32.85208488912081
0.3744646809936967
Tempo de inicio: 0.000000
Tempo de fim: 150654.814377
Tempo de execucao total: 150654.814377

[-30.0, 43.0, 20.0]
[[ 66.]]
[[ 68.96968414]]
[[ -38.01703854]]
[[ -45.]]
44.896039381537435
0.3740822770050727
Tempo de inicio: 0.000000
Tempo de fim: 150655.521579
Tempo de execucao total: 150655.521579

[29.0, 33.0, 20.0]
[[ 66.]]
[[ 159.]]
[[ -50.]]
[[ -90.]]
35.2520944297849
0.37400041200453416
Tempo de inicio: 0.000000
Tempo de fim: 150655.896057
Tempo de execucao total: 150655.896057

[0.0, 41.0, 20.0]
[[ 66.]]
[[ 159.]]
[[ -50.]]
[[ -90.]]
32.85208488912081
0.3743785620028756
Tempo de inicio: 0.000000
Tempo de fim: 150656.270887
Tempo de execucao total: 150656.270887
]

Prompt de Comando
[[386, 36], [104, 82]]
['-38,43,20', '29,33,20']
['0,41,20']
Latencia: 31.0270
Latencia: 30.3585
Latencia: 32.1987
finish

8
Starting
center
(374, 239)
(479, 476)
s
[[248, 47]]
p
p
[[386, 36], [104, 82]]
['-38,43,20', '29,33,20']
['0,41,20']
Latencia: 29.9679
Latencia: 29.9077
Latencia: 32.6018
finish

9
Starting
center
(374, 239)
(479, 476)
s
[[248, 47]]
p
p
p
[[386, 36], [104, 82]]
['-38,43,20', '29,33,20']
['0,41,20']
Latencia: 31.3701
Latencia: 29.5965
Latencia: 30.5506
finish

Tempo de inicio: 0.000000
Tempo de fim: 14.813907
Tempo de execucao: 14.813907
Latencia total: 981.1896

C:\Users\marco\Desktop\TCC20223\SEnew\camera>

```

Figura 16 – Conexão cliente e servidor nuvem híbrida  
 Fonte: elaborado pelo autor

## 5 Apresentação e análise dos resultados

Após a realização das 10 execuções em cada configuração, levando em conta eventuais variações causadas pela utilização da rede *Wi-Fi* e da VPN, quando se aplica, os dados a seguir foram coletados. Os algoritmos utilizados calcularam:

- tempo total de execução do processo;
- latência individual de cada conexão;
- latência total do processo.

Para calcular o tempo de execução total do processo, o algoritmo cliente utiliza a biblioteca *timeit* do Python salvando o tempo de início e fim do mesmo, que subtraindo o tempo final do inicial, resultou na duração total de cada execução. Para calcular a latência individual de cada conexão, cada vez que o algoritmo cliente efetua uma conexão com o servidor e a cada peça encontrada na imagem do espaço de trabalho (são três peças válidas no total), o mesmo calcula o RTT dessa conexão utilizando a biblioteca *verbose\_ping* do Python, realizando assim o cálculo 30 vezes para as 10 execuções. Já a latência total da execução nada mais é do que o somatório de todas as 30 latências individuais de cada conexão ao servidor.

### 5.1 Conexão local

Com base nos resultados das repetidas execuções na conexão local, a Tabela 2 apresenta os tempos totais de execução do processo para cada uma das execuções em segundos.

	Tempo total de execução
<b>Execução 1</b>	14.7496
<b>Execução 2</b>	14.5677
<b>Execução 3</b>	14.8397
<b>Execução 4</b>	14.6272
<b>Execução 5</b>	15.0784
<b>Execução 6</b>	14.5305
<b>Execução 7</b>	14.6505
<b>Execução 8</b>	15.0283
<b>Execução 9</b>	15.4223
<b>Execução 10</b>	15.1703

Tabela 2 – Tempo total de execução do código cliente na conexão local

Calculando a média dos tempos totais de execução do processo para a conexão local apresentados na Tabela 2 temos 14.8665 segundos, a variância de 0.0792 segundos e o desvio padrão igual a 0.2815 segundos. A diferença entre o maior e o menor valor observados é de 0.8917 segundos, ou seja, 6.1374%.

A Tabela 3 apresenta as latências totais de execução do processo para cada uma das execuções em milissegundos.

	Latência total de execução
<b>Execução 1</b>	7.7607
<b>Execução 2</b>	8.0830
<b>Execução 3</b>	7.4078
<b>Execução 4</b>	7.5417
<b>Execução 5</b>	8.4212
<b>Execução 6</b>	7.0316
<b>Execução 7</b>	7.2149
<b>Execução 8</b>	7.4601
<b>Execução 9</b>	7.0377
<b>Execução 10</b>	7.8586

Tabela 3 – Latência total de execução do código cliente na conexão local

O valor médio das latências totais apresentadas da Tabela 3 é de 7.5817 milissegundos, a valor da variância é de 0.1844 milissegundos e o desvio padrão igual a 0.4294 milissegundos. A diferença entre o maior e o menor valor observados é de 1.3896 milissegundos, ou seja, 19.7622%.

O gráfico da Figura 17 exibe as latências individuais de cada conexão realizada para cada um dos três objetos nas 10 execuções do teste na conexão local. Cada cor no referido gráfico representa uma execução com seus 30 conexões, assim podemos ver como a latência varia dentro de cada execução.

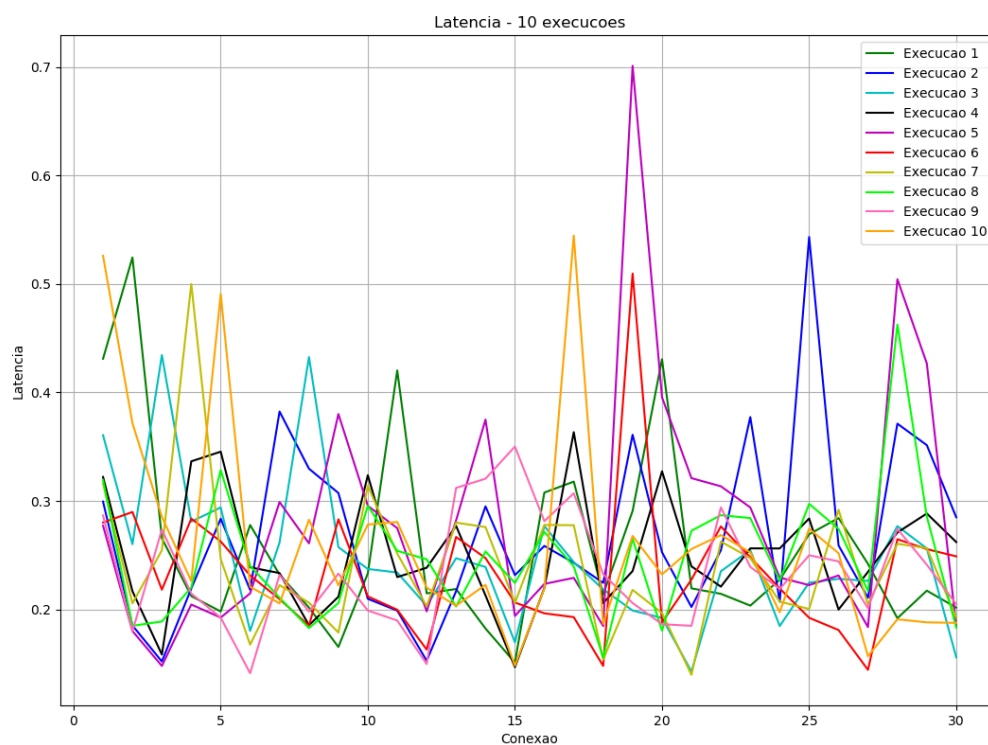


Figura 17 – Variação das latências entre as conexões das 10 execuções na conexão local  
Fonte: elaborado pelo autor

Para cada execução, são realizadas 30 conexões ao servidor, o gráfico da Figura 18 exibe a latência média dessas 30 conexões para cada uma das 10 execuções realizadas.



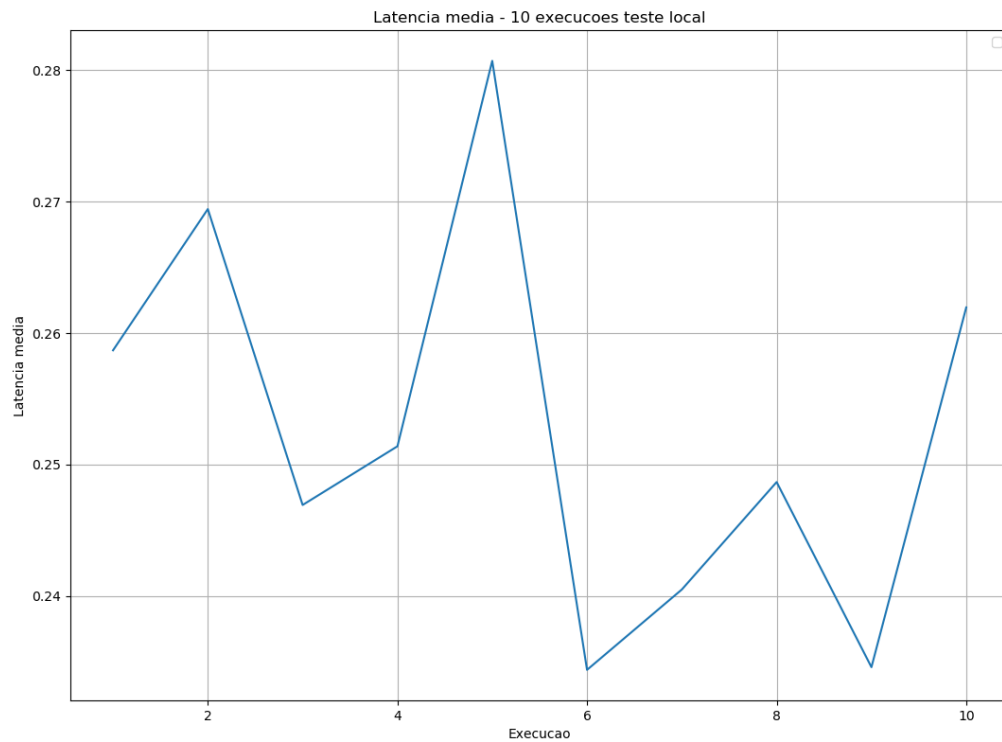


Figura 18 – Valor médio das latências na conexão local  
Fonte: elaborado pelo autor

A média dos valores apresentados no gráfico da Figura 18 é de 0.2527 milissegundos com variância de 0.0002 milissegundos e desvio padrão de 0.0143 milissegundos. A diferença entre o maior e o menor valor médio observados é de 0.0463 milissegundos, ou seja, 19.7622%.

## 5.2 Nuvem Privada

Na sequência de 10 execuções realizadas no *OpenNebula* como nuvem privada, foram obtidos os resultados apresentados a seguir.

A Tabela 4 apresenta os tempos totais de execução do processo na nuvem privada para cada uma das 10 execuções em segundos.

	Tempo total de execução
<b>Execução 1</b>	16.1578
<b>Execução 2</b>	15.9619
<b>Execução 3</b>	16.0560
<b>Execução 4</b>	16.0845
<b>Execução 5</b>	15.9578
<b>Execução 6</b>	15.7158
<b>Execução 7</b>	15.7462
<b>Execução 8</b>	16.0330
<b>Execução 9</b>	16.0713
<b>Execução 10</b>	16.1910

Tabela 4 – Tempo total de execução do código cliente na nuvem privada

A média dos tempos totais de execução do processo na nuvem privada apresentados na Tabela 4 é de 15.9975 segundos, a variância calculada é igual a 0.0226 segundos e o desvio padrão igual a 0.1503 segundos. A diferença entre o maior e o menor valor observados é de 0.4751 segundos, ou seja 3.0237%.

A Tabela 5 mostra a latência total do processo para cada uma das 10 execuções em milissegundos.

	Latência total de execução
<b>Execução 1</b>	141.7602
<b>Execução 2</b>	122.1695
<b>Execução 3</b>	116.5929
<b>Execução 4</b>	161.1061
<b>Execução 5</b>	180.1544
<b>Execução 6</b>	173.7658
<b>Execução 7</b>	111.4374
<b>Execução 8</b>	118.8706
<b>Execução 9</b>	177.1119
<b>Execução 10</b>	151.6159

Tabela 5 – Latência total de execução do código cliente na nuvem privada

O valor médio das latências totais na nuvem privada apresentados na Tabela 5 é de 145.4584 milissegundos, a variância calculada é de 654.3436 milissegundos e o desvio padrão igual a 25.5801 milissegundos. A diferença entre o maior e o menor valor observados é de 68.717 milissegundos, ou seja, 61.6642%.

Para cada execução são realizadas 30 conexões, o gráfico da Figura 19 exibe a variação das latências individuais de cada conexão para cada um dos três objetos, onde cada cor representa uma execução.

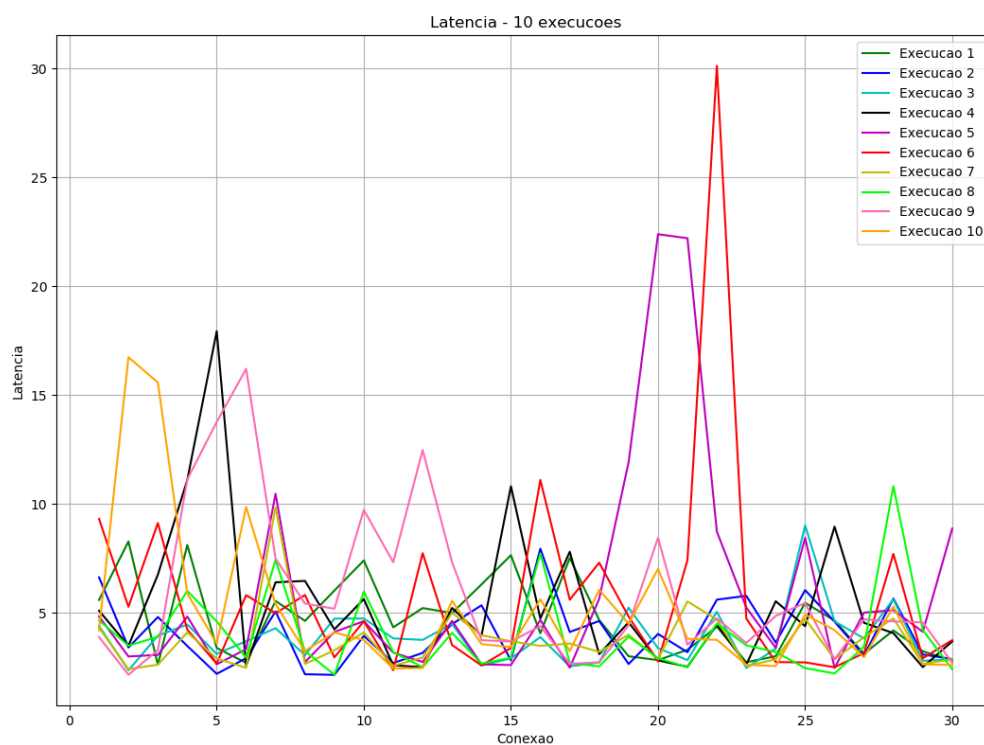


Figura 19 – Variação das latências entre as conexões das 10 execuções na nuvem privada  
Fonte: elaborado pelo autor

Foi calculada a média das latências para cada uma das 30 conexões, o gráfico da Figura 20 mostra a latência média para cada execução. A média dos valores apresentados no gráfico da Figura 20 é de 4.8486 milissegundos, a variância igual a 0.7270 milissegundos e o desvio padrão igual a 0.8526 milissegundos. A diferença entre o maior e o menor valor observados para essa configuração é de 2.2905 milissegundos, ou seja, 61.6642%.

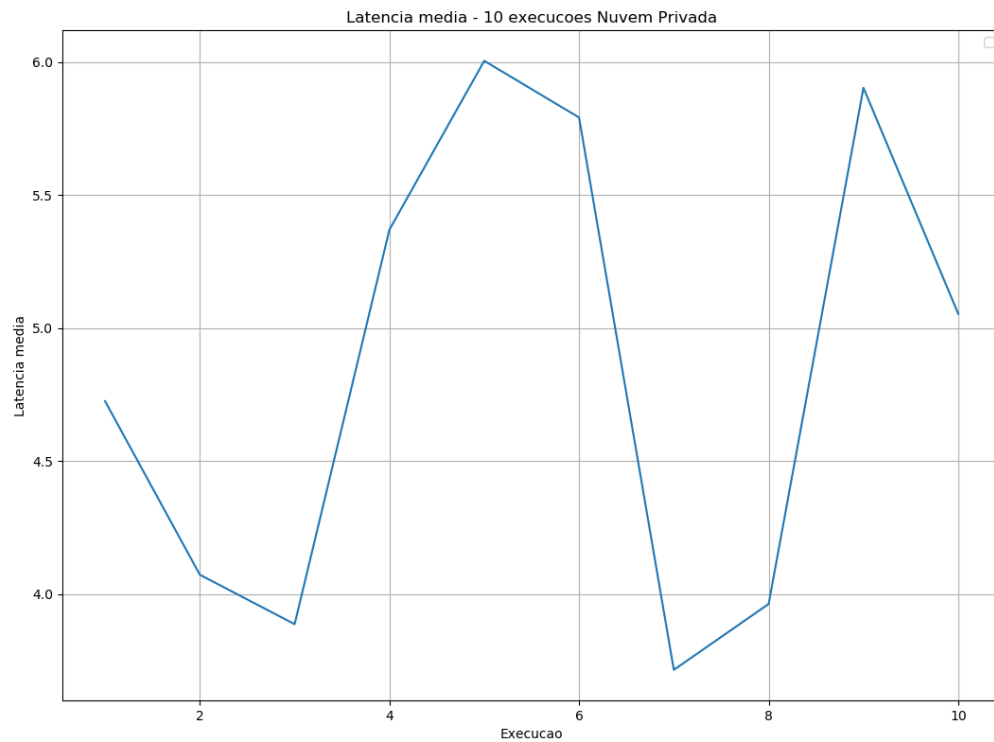


Figura 20 – Valor médio das latências na nuvem privada  
Fonte: elaborado pelo autor

### 5.3 Nuvem Pública

Os dados obtidos nos testes realizados no *Google Cloud* como nuvem pública são exibidos na sequência.

A Tabela 6 mostra, em segundos, a variação dos tempos totais de execução do processo na nuvem pública para cada execução realizada.

	Tempo total de execução
<b>Execução 1</b>	20.4816
<b>Execução 2</b>	19.2213
<b>Execução 3</b>	19.8587
<b>Execução 4</b>	19.7521
<b>Execução 5</b>	20.0707
<b>Execução 6</b>	20.1271
<b>Execução 7</b>	19.4467
<b>Execução 8</b>	19.3637
<b>Execução 9</b>	19.8157
<b>Execução 10</b>	19.7441

Tabela 6 – Tempo total de execução do código cliente na nuvem pública

A média dos tempos totais de execução do processo na nuvem pública mostrados na Tabela 6 é de 19.7882 segundos, a variância calculada é igual a 0.1302 segundos e o desvio padrão igual a 0.3609 segundos. A diferença entre o maior e o menor valor observados é de 1.2602 segundos, ou seja, 4.7124%.

A Tabela 7 mostra as latências totais do processo para cada execução em milissegundos.

	Latência total de execução
<b>Execução 1</b>	5375.2039
<b>Execução 2</b>	5413.9826
<b>Execução 3</b>	5375.1738
<b>Execução 4</b>	5379.0206
<b>Execução 5</b>	5366.4946
<b>Execução 6</b>	5435.5068
<b>Execução 7</b>	5383.4786
<b>Execução 8</b>	5360.5762
<b>Execução 9</b>	5345.0034
<b>Execução 10</b>	5367.2255

Tabela 7 – Latência total de execução do código cliente na nuvem pública

O valor médio das latências totais apresentados na Tabela 7 é de 5380.1666 milissegundos, a variância calculada é de 624.2528 milissegundos e o desvio padrão igual a 24.9850 milissegundos. A diferença entre o maior e o menor valor observados é de 90.5034 milissegundos, ou seja, 1.6932%.

Para cada execução realizada, o algoritmo gera 30 valores de latências individuais, o gráfico da Figura 21 exibe os valores das latências individuais para cada conexão realizada nas 10 execuções do teste na nuvem pública, onde cada cor representa uma execução e suas variações.

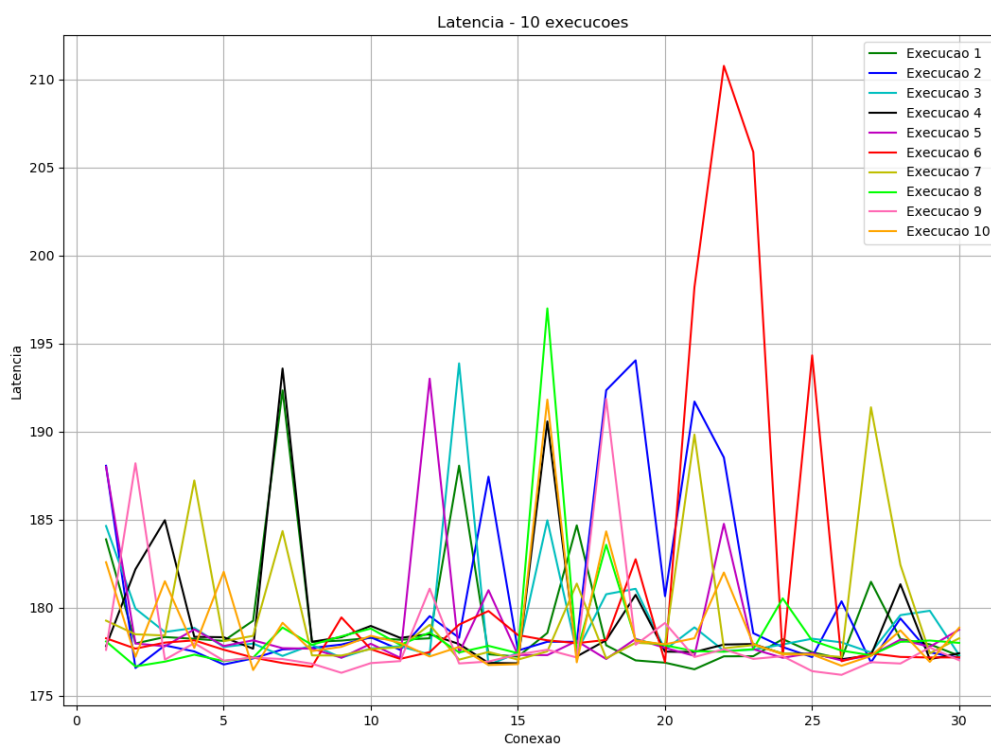


Figura 21 – Variação das latências entre as conexões das 10 execuções na nuvem pública  
Fonte: elaborado pelo autor

O gráfico da Figura 22 mostra a latência média das 30 conexões realizadas em cada execução do teste. A média dos valores apresentados no gráfico da Figura 22 é de 179.3388 milissegundos, com variância de 0.6936 milissegundos e desvio padrão de 0.8328 milissegundos. A diferença entre o maior e o menor valor observados é de 3.0167 milissegundos, ou seja, 1.6932%.

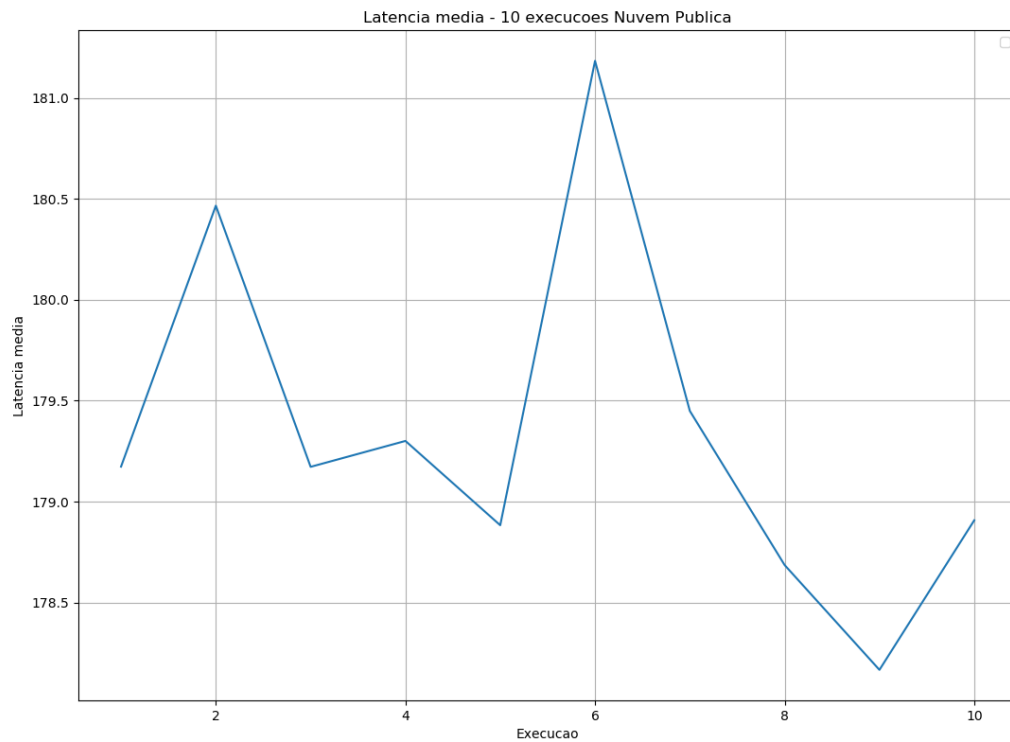


Figura 22 – Valor médio das latências na nuvem pública  
Fonte: elaborado pelo autor

## 5.4 Nuvem Híbrida

Realizadas todas as execuções no *OpenNebula* através da VPN da UFOP, os dados a seguir foram observados.

A Tabela 8 apresenta os tempos totais de execução do processo na nuvem híbrida para cada uma das 10 execuções. Nos gráficos e tabelas apresentados nesta seção podemos observar uma variação anormal na execução de número 6 que pode ser justificada como uma falha de conexão ou variação na conexão *Wi-Fi* ou VPN.

	Tempo total de execução
<b>Execução 1</b>	14.7616
<b>Execução 2</b>	14.8375
<b>Execução 3</b>	14.8279
<b>Execução 4</b>	14.7303
<b>Execução 5</b>	14.8689
<b>Execução 6</b>	15.3391
<b>Execução 7</b>	14.6171
<b>Execução 8</b>	14.7175
<b>Execução 9</b>	14.6050
<b>Execução 10</b>	14.7570

Tabela 8 – Tempo total de execução do código cliente na nuvem híbrida

A média dos tempos totais de execução do processo na nuvem híbrida apresentados na Tabela 8 é de 14.8062 segundos, a variância calculada é de 0.0383 segundos e o desvio padrão igual a 0.1958 segundos. A diferença entre o maior e o menor valor observados é de 0.7341 segundos, ou seja, 5.0263%.

A Tabela 9 mostra as latências totais do processo em cada uma das 10 execuções em milissegundos.

	Latência total de execução
<b>Execução 1</b>	996.1878
<b>Execução 2</b>	1014.9991
<b>Execução 3</b>	946.0537
<b>Execução 4</b>	891.5355
<b>Execução 5</b>	960.4198
<b>Execução 6</b>	1379.5773
<b>Execução 7</b>	903.0129
<b>Execução 8</b>	944.4503
<b>Execução 9</b>	942.8352
<b>Execução 10</b>	1021.0225

Tabela 9 – Latência total de execução do código cliente na nuvem híbrida

O valor médio das latências totais na nuvem híbrida mostrados na Tabela 9 é de 1000.0094 milissegundos, a variância igual a 17676.1801 milissegundos e o desvio padrão igual a 132.9517 milissegundos. A diferença entre o maior e o menor valor observados é de 488.0418 milissegundos, ou seja, 54.7417%.

O gráfico da Figura 23 exibe a variação das 30 latências individuais registradas para cada conexão realizada em cada uma das 10 execuções do teste, onde cada cor representa uma execução e as variações que as mesmas apresentaram.



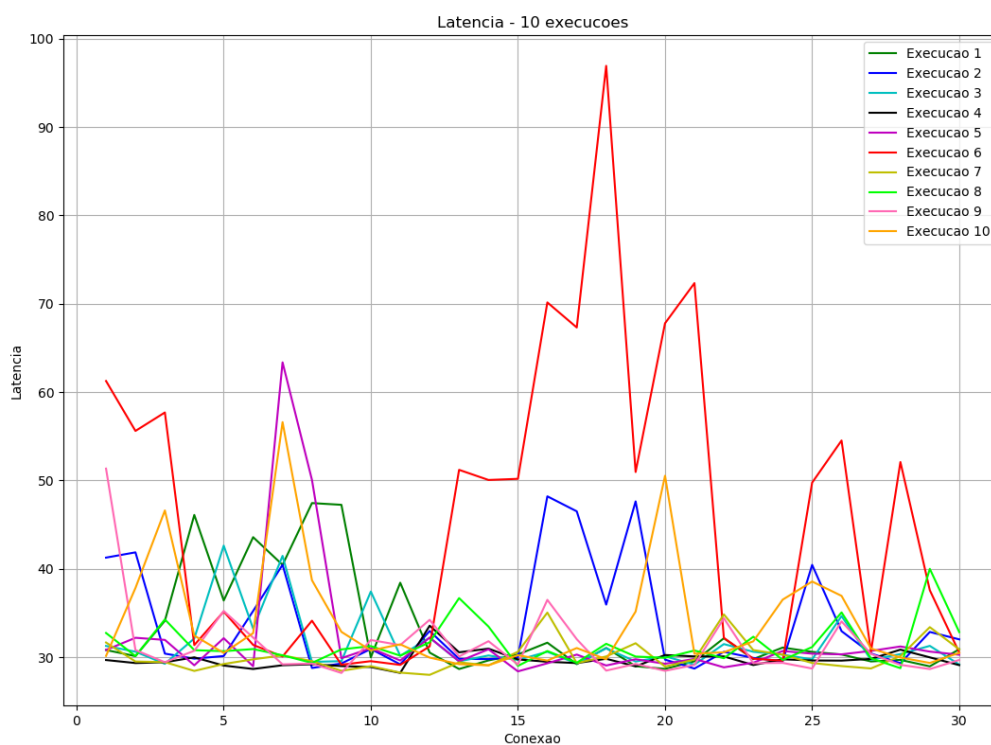


Figura 23 – Variação das latências entre as conexões das 10 execuções na nuvem híbrida  
Fonte: elaborado pelo autor

O gráfico da Figura 24 apresenta a média das 30 latências registradas para cada uma das 10 execuções. A média dos valores apresentados no gráfico da Figura 24 é de 33.3336 milissegundos, a variância calculada é de 19.6402 milissegundos e o desvio padrão igual a 4.4317 milissegundos. A diferença entre o maior e o menor valor observados é de 16.2680 milissegundos, ou seja, 54.7417%.

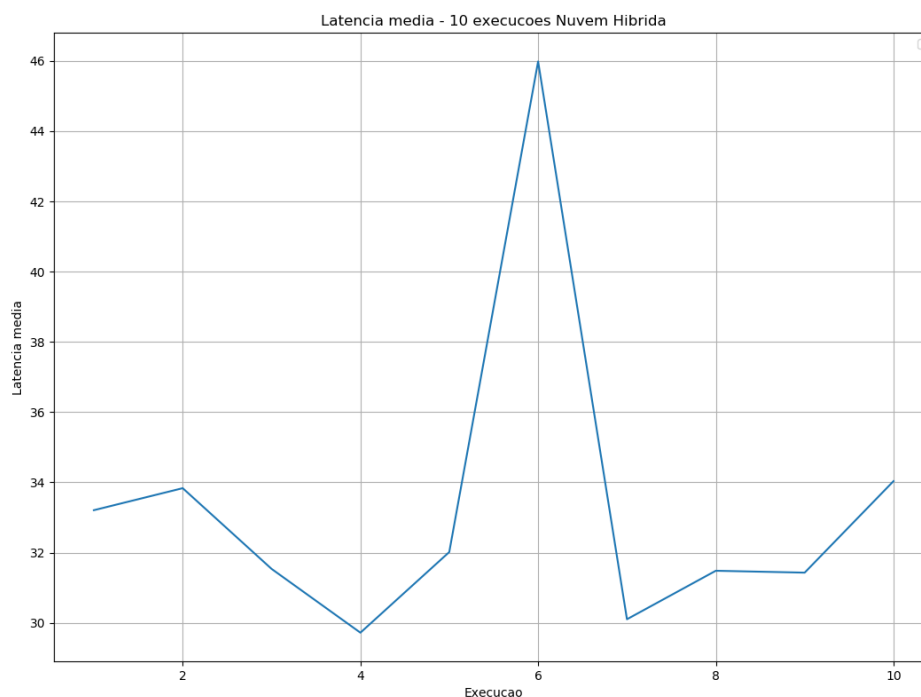


Figura 24 – Valor médio das latências na nuvem híbrida

Fonte: elaborado pelo autor

## 5.5 Análise dos Resultados

Analisando as quatro configurações de testes realizados, podemos observar que as médias dos tempos totais de execução mostrados no gráfico da Figura 25 para a conexão local e para a nuvem híbrida diferiram pouco, logo depois vem a nuvem privada e exigindo maior tempo de processamento, a nuvem pública. Os valores exatos de cada média são mostradas na Tabela 10.

	Tempo de execução
Conexão local	14.8665
Nuvem híbrida	14.8062
Nuvem privada	15.9975
Nuvem pública	19.7882

Tabela 10 – Média tempo de execução

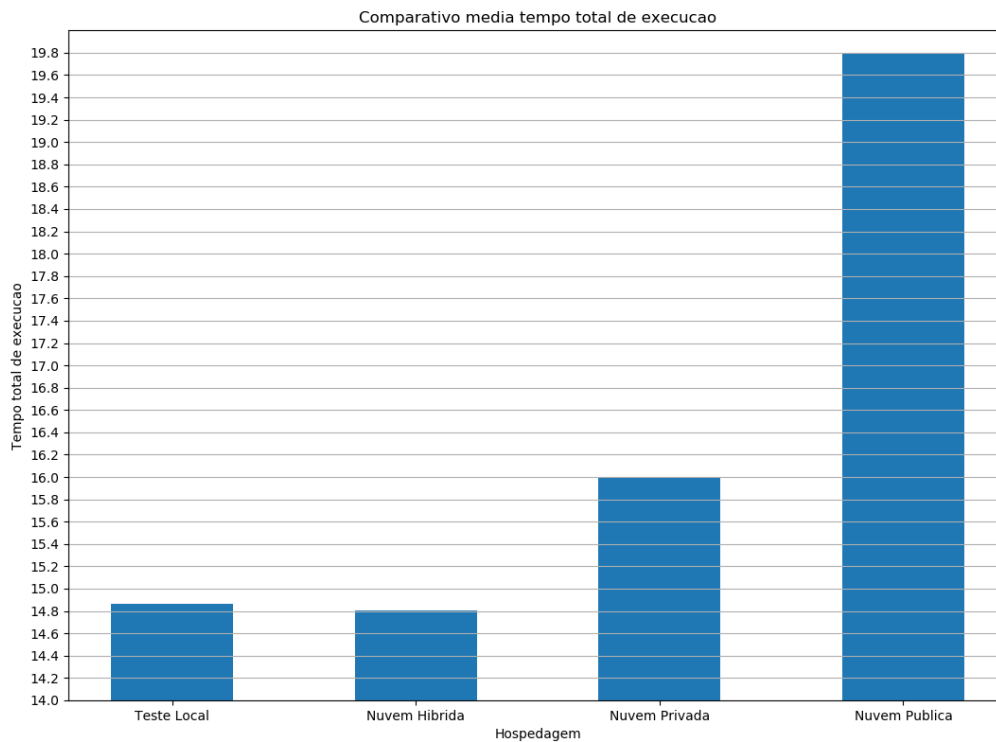


Figura 25 – Tempo médio de execução por configuração

Fonte: elaborado pelo autor

O tempo de execução está diretamente relacionado ao processador da máquina, sendo assim, o maior tempo total de execução na nuvem pública ocorre devido ao fato de os recursos serem divididos entre usuários do servidor e a limitação do processador utilizado na configuração da máquina virtual na nuvem. Já a nuvem híbrida tem resultados muito próximos dos obtidos na conexão local (que quase não tem limitações de processamento) devido ao fato de associar qualidades da nuvem pública com qualidades da nuvem privada, trazendo resultados otimizados para o teste. Os tempos de execução das nuvens privada e híbrida ficaram muito próximos, variando apenas 1,1913 segundos, ou seja, 8.0459%.

Os valores médios de latência por conexão das quatro configurações são descritos na Tabela 11. Com relação aos valores obtidos de latência de cada conexão, observa-se que, como esperado, a conexão local tem valor de latência quase nulo, já que o teste é feito localmente. A nuvem privada é a que mostra melhor resultado, já que seus recursos são exclusivos por usuário, não havendo compartilhamento dos mesmos, ao contrário da nuvem pública que apresenta maior latência devido ao fato de seus recursos serem compartilhados entre usuários do serviço.

	Latência por conexão
Conexão local	0.2527
Nuvem híbrida	33.3336
Nuvem privada	4.8486
Nuvem pública	179.3388

Tabela 11 – Média da latência por conexão

O gráfico da Figura 26 apresenta as variações das latências médias para cada uma das 10 execuções em cada configuração de serviço testado. Observa-se uma pequena variação da latência para cada conexão em cada uma das 10 execuções, porém, comparando-as, vemos nitidamente que a nuvem privada se destaca mais uma vez, comprovando sua eficácia nos testes realizados. Podemos observar também a variação anormal citada anteriormente na execução de número 6 para o serviço de nuvem híbrida, que pode ser justificado como falha de conexão ou variação na conexão *Wi-Fi* ou VPN, excluindo esse detalhe, as variações observadas são consideradas normais.

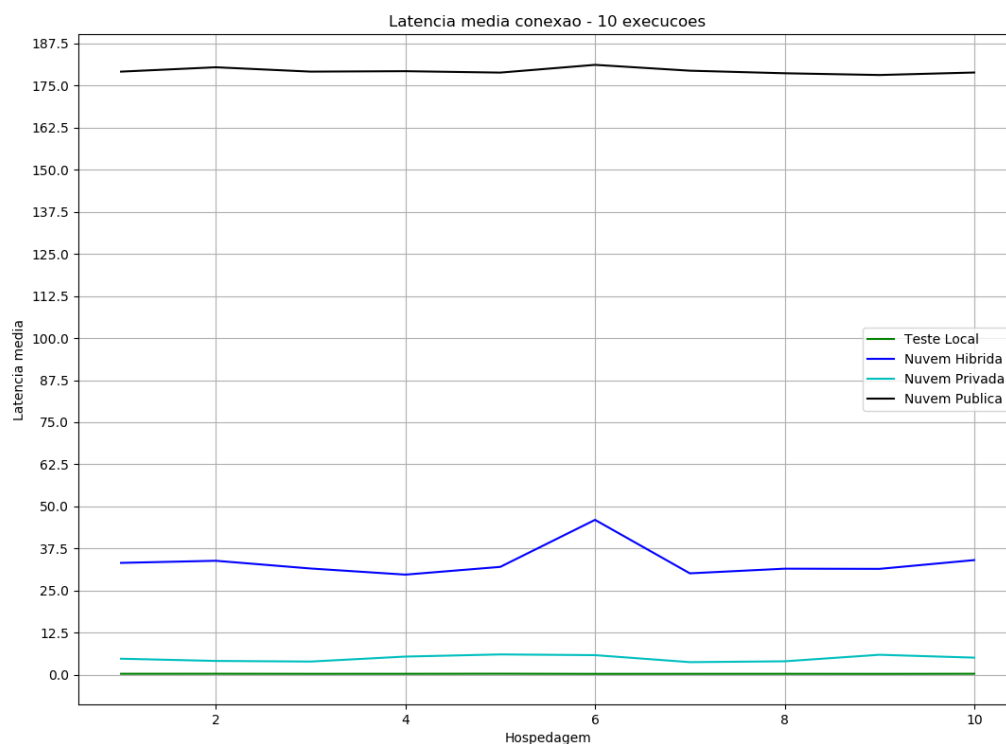


Figura 26 – Latência média da conexão por configuração

Fonte: elaborado pelo autor

A Tabela 12 mostra as latências totais médias para cada configuração em milissegundos. As latências totais do processo, tais como as latências individuais de cada conexão realizada durante o mesmo, variaram muito entre as configurações. Como dito

anteriormente, para o teste local, a latência é quase nula. Entre os serviços de Computação em Nuvem trabalhados, observa-se que a nuvem privada obteve menor latência. A nuvem híbrida que demanda conexão com VPN vem em terceiro lugar e por último a nuvem pública, que teve sua latência cinco vezes maior que da nuvem híbrida, mais uma vez devido ao fato de não priorizar recursos e menos ainda a velocidade de comunicação, já que diversos usuários compartilham dos mesmos recursos.

	Latência total
Conexão local	7.5817
Nuvem híbrida	1000.0094
Nuvem privada	145.4584
Nuvem pública	5380.1666

Tabela 12 – Média da latência total

## 6 Conclusão

A aplicação da Computação em Nuvem em um cenário industrial é uma realidade atual. Qualquer empresa de pequeno ou grande porte pode usufruir das vantagens dessa ferramenta que é um item importante da chamada Indústria 4.0. Com base nessa importância, foram analisadas características de conexão e rede no serviço de Computação em Nuvem na Indústria 4.0, realizando assim testes simulando tal cenário e comparações entre os resultados obtidos.

A TI e seus assuntos relacionados estão em constante evolução em pesquisas e aplicações. O mundo está cada vez mais conectado e se comunicando através de *gadgets* que coletam e compartilham dados de seus usuários e os enviam para nuvem onde o *big data* os processa e analisa. Vidas podem ser salvas, dados sobre o sono e a saúde de seus usuários são acompanhados de forma mais compreensível, análise e previsões sobre economia, política e esportes são realizadas, as pessoas estão cada vez mais dependentes da tecnologia e a utilizam para diversos fins.

Conclui-se que, a nuvem pública, como descrito neste trabalho, é o tipo mais simples de nuvem, onde os recursos e a segurança não são a prioridade. O tempo de execução não é otimizado para o processo, aumentando consideravelmente para essa configuração, podendo ser justificado pelo compartilhamento de recursos entre usuários e limitações nos mesmos e a devida melhoria dessas configurações acarreta no aumento do preço do serviço final. A nuvem privada tem um bom tempo de processamento total, diferindo em apenas dois segundos da nuvem híbrida. A nuvem híbrida apresenta o melhor tempo de execução total devido ao uso da VPN que escolhe a melhor rota para chegar ao destino que como o teste local, apenas gastou tempo de processamento, onde a latência foi praticamente nula.

Para tal cenário proposto, a nuvem privada se mostra a melhor opção, como o tempo de processamento não diferiu tanto do menor valor obtido, mas a latência se tornou o ponto de maior criticidade e importância entre as três principais configurações, esta se destacou nos teste realizados, porém, vale ressaltar que a escolha de cada serviço em nuvem depende muito da aplicação, podendo assim variar os recursos adquiridos com base na mesma.

### 6.1 Limitações e trabalhos futuros

Como sugestão para trabalhos futuros, pode-se realizar testes em outras características da Computação em Nuvem empregada na Indústria 4.0 como segurança, processamento, gravação e leitura de informações, bem como testes em outros equipamentos inteligentes

empregados na indústria.

# Referências

- ALBERTIN, M. R. et al. Principais inovações tecnológicas da indústria 4.0 e suas aplicações e implicações na manufatura. 2017. Citado na página 21.
- CONTRIBUTOR. *Virtual Machine Manager*. 2022. Url,<https://virt-manager.org/>. Citado na página 32.
- DEBIAN. 2022. Url,<https://www.debian.org/releases/stretch/debian-installer/>. Citado na página 32.
- HERMANN, M.; PENTEK, T.; OTTO, B. Design principles for industrie 4.0 scenarios. In: IEEE. *System Sciences (HICSS)*, 2016 49th Hawaii International Conference on. [S.l.], 2016. p. 3928–3937. Citado na página 21.
- INOUE, J. et al. Indústria 4.0–impactos da tecnologia da informação na nova indústria. *Pesquisa e Ação (ISSN 2447-0627)*, v. 5, p. N1, 2019. Citado 2 vezes nas páginas 21 e 25.
- LLC, G. *Google Cloud*. 2023. Url,<https://cloud.google.com/?hl=pt-br>. Citado na página 35.
- OLIVEIRA, M. H. d. L. Cloud test: melhoria da confiabilidade de testes mobile com a utilização de computação em nuvem. 2021. Citado na página 27.
- PAZ, A. C. M.; LOOS, M. J. A importância da computação em nuvem para a indústria 4.0. *Revista Gestão Industrial*, v. 16, n. 2, 2020. Citado 2 vezes nas páginas 16 e 24.
- PEDROSA, P. H.; NOGUEIRA, T. Computação em nuvem. *artigo disponível em http://www.ic.unicamp.br/~ducatte/mo401/1s2011*, v. 2, 2011. Citado 3 vezes nas páginas 20, 25 e 26.
- POSSOBOM, C. C. Estudo de caso: cloud computing-computação em nuvem. 2012. Citado na página 20.
- RODRIGUES, E. M. Controle de um braço robótico articulado com 6 graus de liberdade. 2022. Citado 7 vezes nas páginas 11, 12, 22, 23, 24, 28 e 30.
- SEGALIN, D. *DETECÇÃO INTELIGENTE DE SURTOS DE PROCESSAMENTO NA COMPUTAÇÃO EM NUVEM*. Tese (Doutorado) — Pontifícia Universidade Católica do Paraná, 2013. Citado na página 26.
- SILVA, L. P. O. da et al. Gerência de recursos em sistemas distribuídos para cloud gaming. 2020. Citado na página 26.
- SYSTEMS, O. *OpenNebula 5.6 Documentation*. 2022. Url,<https://docs.opennebula.io/5.6/>.. Citado na página 32.
- VINICIUS, M. A. et al. Indústria 4.0. *Revista Pesquisa e Ação*, v. 5, n. 1, p. 127–147, 2019. Citado na página 16.
- VOLUNTEERS. *Putty.org*. 2023. Url,<https://www.putty.org/>. Citado na página 37.



---

YUE, X. et al. Cloud-assisted industrial cyber-physical systems: An insight. *Microprocessors and Microsystems*, Elsevier, v. 39, n. 8, p. 1262–1270, 2015. Citado na página [20](#).