



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE CRATEÚS
CURSO DE CIÊNCIA DA COMPUTAÇÃO

NATÁ SANTANA DE MORAIS

**AHPSELECTION: APLICAÇÃO DA TÉCNICA AHP NO PROBLEMA DE SELEÇÃO
DE MÁQUINAS VIRTUAIS PARA MIGRAÇÃO EM UM AMBIENTE DE DATA
CENTER**

CRATEÚS

2019

NATÃ SANTANA DE MORAIS

AHPSELECTION: APLICAÇÃO DA TÉCNICA AHP NO PROBLEMA DE SELEÇÃO DE
MÁQUINAS VIRTUAIS PARA MIGRAÇÃO EM UM AMBIENTE DE DATA CENTER

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Campus de Crateús da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Ciência da Computação.

Orientador: Prof. Me. Filipe Fernandes
dos Santos Brasil de Matos

CRATEÚS

2019

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Biblioteca Universitária
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

M826a Morais, Natã Santana de.
AHPSelection: Aplicação da Técnica AHP no Problema de Seleção de Máquinas Virtuais para Migração em um ambiente de Data center / Natã Santana de Morais. – 2019.
82 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Crateús, Curso de Ciência da Computação, Crateús, 2019.
Orientação: Prof. Me. Filipe Fernandes dos Santos Brasil de Matos.

1. Política de Seleção de Máquinas Virtuais. 2. AHP. 3. Computação em Nuvem. I. Título.

CDD 004

NATÃ SANTANA DE MORAIS

AHPSELECTION: APLICAÇÃO DA TÉCNICA AHP NO PROBLEMA DE SELEÇÃO DE
MÁQUINAS VIRTUAIS PARA MIGRAÇÃO EM UM AMBIENTE DE DATA CENTER

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
do Campus de Crateús da Universidade Federal
do Ceará, como requisito parcial à obtenção do
grau de bacharel em Ciência da Computação.

Aprovada em:

BANCA EXAMINADORA

Prof. Me. Filipe Fernandes dos Santos Brasil de
Matos (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Me. Lisieux Marie Marinho dos Santos Andrade
Universidade Federal do Ceará (UFC)

Prof. Me. Francisco Anderson de Almada Gomes
Universidade Federal do Ceará (UFC)

Dedico este trabalho primeiramente a Deus, por ser essencial em minha vida e meu guia. Aos meus pais, meu irmão, minha avó e ao amor da minha vida que, com muito carinho e apoio, não mediram esforços para que eu chegasse até esta etapa de minha vida.

AGRADECIMENTOS

Primeiramente a Deus que permitiu que tudo isso acontecesse e pela sua misericórdia em minha vida. E mesmo nos momentos difíceis, ele foi meu guia.

Aos meus pais, meu irmão, minha vó e o amor da minha vida, pelo incentivo e pela ajuda nos momentos de dificuldade e desânimo. Por não desistirem de mim e não cansarem de dizer que daria tudo certo, mesmo quando tudo parecia que iria fracassar.

Ao Prof. Me. Filipe Fernandes dos Santos Brasil de Matos por aceitar me orientar neste trabalho e ser um ótimo exemplo de organização, trabalho e resultado.

À Prof. Me. Lisieux Marie Marinho dos Santos Andrade pelas suas sugestões e ideias sobre o método de tomada de decisão neste trabalho.

Ao Prof. Me. Francisco Anderson de Almada Gomes pelas suas sugestões de organização nesse trabalho.

Ao Prof. Me. Roberto Cabral Rabelo Filho pela oportunidade de fazer a disciplina de criptografia, na qual me saltou os olhos e me fez gostar mais ainda de segurança, principalmente no tráfego em redes.

À Prof. Me. Lílian de Oliveira Carneiro pelo seu exemplo na área da matemática e sua contribuição em meu aprendizado e pela colaboração em transformar a matemática, uma paixão.

Ao Prof. Dr. Giannini Italino Alves Vieira pela oportunidade de fazer me apaixonar mais ainda por matemática através de seu ensino espetacular.

Aos amigos que fiz, tanto no início como no final da graduação e aos amigos que já fazem parte da minha vida que sempre me apoiaram e fizeram os meus dias mais produtivos e engraçados.

“Combati o bom combate, acabei a carreira, guardei a fé.”

(2 Timóteo 4:7)

RESUMO

A Computação em Nuvem é um novo modelo computacional que revolucionou a área de Tecnologia da Informação (TI). Tal paradigma permite a oferta e o consumo sob demanda de recursos computacionais presentes em grandes infraestruturas conhecidas como *data centers* sob o formato de serviços. O sucesso desse modelo motivou um aumento na quantidade de *data centers*, o que fez o consumo de energia de tais infraestruturas chegarem a níveis preocupantes. Um das formas de minimizar este consumo de energia é através de uma distribuição inteligente de Máquinas Virtuais (VMs) entre os servidores que o compõe. Tal distribuição compreende o uso de Políticas de Alocação na escolha do melhor servidor para receber uma VM e de uma Política de Seleção na escolha de uma VM para alocação ou migração. Este trabalho, apresenta uma nova Política de Seleção de VMs, a AHPSelection, que mapeia o processo de escolha de uma VM para migração como um problema de tomada de decisão e aplica a técnica Analytic Hierarchy Process (AHP) como ferramenta para auxiliar neste procedimento. Em um primeiro momento, a AHPSelection apresentou resultados intermediários em termos de desempenho e de consumo de energia. Tal comportamento motivou a criação de uma versão modificada da AHPSelection (chamada de AHPSelection2) que apresentou bons resultados em termos de consumo de energia e de quantidade de migrações em cenários onde há pouca variação na demanda por recursos por parte das VMs. A AHPSelection2 conseguiu reduzir de quatro a cinco vezes o número total de migrações em alguns cenários, bem como proporcionou uma economia de energia média de 15 Kwh em outro.

Palavras-chave: Computação em Nuvem. *Data center*. Política de Seleção. Máquinas Virtuais. AHP

ABSTRACT

Cloud Computing is a new computational model that has revolutionized the area of Information Technology (IT). Such a paradigm allows the on-demand supply and consumption of computational resources present in large infrastructures known as data centers in the form of services. The success of this model has led to an increase in the number of data centers, which has led to the energy consumption of such infrastructures reaching worrying levels. One way to minimize this power consumption is by intelligently distributing Virtual Machines (VMs) among its component servers. Such distribution comprises the use of Allocation Policies in choosing the best server to receive a VM and a Selection Policy in choosing a VM for allocation or migration. This paper presents a new VM Selection Policy, AHPSelection, which maps the process of choosing a VM for migration as a decision making problem and applies the Analytic Hierarchy Process (AHP) technique as a tool to assist in this procedure. Initially, AHPSelection presented intermediate results in terms of performance and energy consumption. This behavior led to the creation of a modified version of AHPSelection (called AHPSelection2) that showed good results in terms of power consumption and number of migrations in scenarios where there is little variation in VM resource demand. AHPSelection2 was able to reduce by four to five times the total number of migrations in some scenarios, as well as providing an average energy savings of 15 Kwh in others.

Keywords: Cloud computing. *Data center*. Virtual Machine Selection Policies. AHP

LISTA DE ILUSTRAÇÕES

Figura 1 – Principais tecnologias utilizadas na Computação em Nuvem	20
Figura 2 – Serviços oferecidos pela Computação em Nuvem	22
Figura 3 – Estrutura tradicional de virtualização	24
Figura 4 – Esquema do funcionamento da política de seleção e alocação	27
Figura 5 – Esquema de hierarquia AHP	31
Figura 6 – Índice randômico	33
Figura 7 – Hierarquia para o problema	36
Figura 8 – Matriz de comparação final	38
Figura 9 – Fluxograma da política de seleção VBalance	43
Figura 10 – Hierarquia AHP para o problema de seleção de máquinas virtuais	51
Figura 11 – Consumo de energia do trace PlanetLab	64
Figura 12 – Quantidade de migrações do trace PlanetLab	65
Figura 13 – PDM do trace PlanetLab	66
Figura 14 – SLATAH do trace PlanetLab	67
Figura 15 – Consumo de energia do trace BitBrains	69
Figura 16 – Quantidade de migrações do trace BitBrains	69
Figura 17 – PDM do trace BitBrains	70
Figura 18 – SLATAH do trace BitBrains	71
Figura 19 – Consumo de energia do trace GoogleCluster	72
Figura 20 – Quantidade de migrações do trace GoogleCluser	73
Figura 21 – SLATAH do trace GoogleCluster	74

LISTA DE TABELAS

Tabela 1 – Escala fundamental de números absolutos	32
Tabela 2 – Matriz de comparação dos critérios	36
Tabela 3 – Matriz de comparação dos critérios com a coluna de Prioridade	37
Tabela 4 – Matriz de comparação dos subcritérios	37
Tabela 5 – Matriz de comparação das alternativas com base no critério de Potencial de Salário	38
Tabela 6 – Especificação dos servidores utilizados	56
Tabela 7 – Consumo de energia por carga de trabalho	57
Tabela 8 – Especificação das Máquinas Virtuais	57
Tabela 9 – Especificação das Máquinas Virtuais	58
Tabela 10 – Políticas de Alocação e Seleção adotadas	63

LISTA DE QUADROS

Quadro 1 – Tabela que esquematiza a matriz de comparação dos critérios	52
Quadro 2 – Tabela que esquematiza a matriz de comparação dos subcritérios do critério de Utilização do Processador	53

SUMÁRIO

1	INTRODUÇÃO	14
1.1	Justificativa	16
1.2	Objetivos	17
1.2.1	<i>Objetivos específicos</i>	17
1.3	Organização do Trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	Computação em Nuvem	19
2.2	Virtualização	22
2.3	Computação em Nuvem Verde	24
2.4	Políticas de Alocação e Seleção	26
2.5	Pesquisa Operacional	27
2.5.1	<i>Problema de Programação Linear</i>	28
2.5.2	<i>Representação de um Problema de Programação Linear</i>	28
2.5.3	<i>Técnicas para resolução de PPL</i>	28
2.6	AHP - Analytic Hierarchy Process	29
2.6.1	<i>Índice de Consistência (IC) e Razão de Consistência (RC)</i>	31
2.6.2	<i>Média Geométrica</i>	33
2.6.3	<i>Normalização</i>	34
2.7	AHP através de um exemplo	34
3	TRABALHOS RELACIONADOS	41
3.1	<i>VBalance: A Selection Policy of Virtual Machines for Load Balancing in Cloud Computing</i>	41
3.2	<i>Energy-efficient and SLA-aware Virtual Machine Selection Algorithm for Dynamic Resource Allocation in Cloud Data Centers</i>	43
3.3	<i>End-to-End informed VM Selection in Compute Clouds</i>	45
3.4	<i>Dynamic VM consolidation for energy-aware and SLA violation reduction in cloud computing</i>	47
3.5	Fechamento do Capítulo	48
4	AHPSELECTION	50
4.1	AHP e a Seleção de Máquinas Virtuais	50

4.2	Matrizes de Comparação	52
4.2.1	<i>Algoritmo para a ponderação das alternativas</i>	53
5	AMBIENTE DE TESTES	56
5.1	Configuração Geral do Cenário	56
5.1.1	<i>Cenário (proposto por (MANN; SZABÓ, 2017))</i>	58
5.2	Métricas	58
5.2.1	<i>Consumo de energia</i>	59
5.2.2	<i>Quantidade de migrações</i>	59
5.2.3	<i>Performace Degradation due to Migration</i>	59
5.2.4	<i>SLA Violation Time per Active Host</i>	60
5.3	Políticas de Seleção e Alocação	60
5.4	Configuração do ambiente de teste	61
6	RESULTADOS	63
6.1	<i>Trace PlanetLab</i>	64
6.1.1	<i>Consumo de energia</i>	64
6.1.2	<i>Quantidade de migrações</i>	65
6.1.3	<i>PDM</i>	66
6.1.4	<i>SLATAH</i>	67
6.2	<i>Trace BitBrains</i>	68
6.2.1	<i>Consumo de energia</i>	68
6.2.2	<i>Quantidade de migrações</i>	69
6.2.3	<i>PDM</i>	70
6.2.4	<i>SLATAH</i>	70
6.3	<i>Trace GoogleCluster</i>	72
6.3.1	<i>Consumo de energia</i>	72
6.3.2	<i>Quantidade de migrações</i>	73
6.3.3	<i>PDM</i>	74
6.3.4	<i>SLATAH</i>	74
7	CONCLUSÃO	76
	REFERÊNCIAS	78

1 INTRODUÇÃO

A Internet foi uma revolução e é atualmente a base tecnológica para a Era da Informação (CASTELLS, 2003). Cada vez mais pessoas, grupos e empresas buscam soluções computacionais através da Internet a fim de resolver problemas corriqueiros. As inúmeras formas de interação dos usuários entre si e destes com as informações e serviços oferecidos fizeram surgir alguns modelos arquiteturais de como estruturar as aplicações hospedadas ou que usam a Grande Rede de Computadores chamada a internet.

A Internet está fundamentada na arquitetura Cliente-Servidor (COSTA, 2008). Nesta arquitetura, há dois tipos de máquinas: as clientes e as servidoras. As máquinas servidoras são equipamentos com alto poder de processamento e armazenamento que fornecem algum tipo de recurso ou serviço as demais, por exemplo, uma página, um sistema Web ou um banco de dados. As máquinas clientes, por sua vez, desempenham o papel de requisitar e consumir tais recursos ou serviços. Comumente, neste modelo, os servidores são alocados em *data centers*. Em um *data center*, as máquinas servidoras interagem entre si através de uma rede de alta velocidade e disponibilizam seus recursos e serviços através da Internet para milhões de clientes espalhados pelo mundo.

O modelo tradicional, todavia, apresenta alguns problemas pertinentes para clientes e, principalmente, para empresas mantenedoras dos *data centers*. Os custos operacionais, estes relacionados a implantação, a manutenção e a contínua utilização do *data center*, são responsáveis de 45% a 49% do total dos custos de um *data center* (PONEMON INSTITUTE, 2016). Assim, um dos principais problemas para as empresas são os custos para a criação e a manutenção de um *data center* (KOOMEY *et al.*, 2007). Ao criar um *data center*, a empresa pode superestimar ou subestimar o nível de procura e utilização dos seus recursos. Em ambos os cenários, isso pode representar prejuízo a ela. No caso da superestimativa, a empresa pode ter altos gastos para comprar e manter equipamentos que raramente são utilizados. Já um *data center* subestimado pode prejudicar a qualidade e a disponibilidade dos recursos e serviços ofertados e afastar possíveis clientes.

A Computação em Nuvem foi desenvolvida visando mitigar alguns desses problemas do modelo tradicional (ALJABRE, 2012). O modelo de Computação em Nuvem, pode ser definido como um paradigma ou plataforma que permite acesso a recursos físicos e virtuais tendo como meio de acesso a esses recursos a Internet e administrado sob demanda (SANTOS, 2018). Na Nuvem, os recursos de Tecnologia da Informação (*software e hardware*) são oferecidos

em forma de serviço. Através da Internet, assim como acontece com serviços de telefonia e distribuição de energia elétrica, água e esgoto, por exemplo, os usuários pagam somente por aquilo que foi utilizado por eles. Esse mecanismo é conhecido como *pay-per-use*, que, em tradução livre, é o ato de “pagar pelo o que se usa”. O *pay-per-use* é um modelo de negócio de pagamento por uso, onde os consumidores pagam por unidade de serviço (BOCKEN *et al.*, 2018).

Na Computação em Nuvem, de um modo geral, existem dois tipos de papéis: os Clientes ou Consumidores e os Provedores de serviço. Os Provedores são normalmente empresas que oferecem serviços através da Nuvem, enquanto os Consumidores são pessoas comuns ou, até mesmo, outras empresas que utilizam tais serviços. A prestação de um serviço de Computação em Nuvem requer um certo nível de confiabilidade na relação entre Provedores e Clientes. Para isso, existem os SLAs ou *Service Level Agreements*. O SLA é um contrato estabelecido entre a entidade que fornece o serviço e o cliente que deseja utilizá-lo (PATEL *et al.*, 2009). Neste documento, constam critérios de desempenho relacionados a Qualidade de Serviço (QoS), os direitos e as responsabilidades de ambas as partes na prestação do serviço, assim como possíveis penalidades caso alguma das regras não for obedecida.

A Computação em Nuvem, infelizmente, não apresenta só vantagens. Existem alguns problemas quando se investe neste tipo de modelo, principalmente para os Provedores de serviço. Um destes, é o desafio da redução do consumo de energia pelos *data centers* (SOHRABI *et al.*, 2018). Em um *data center* é preciso que haja confiabilidade e disponibilidade, porém isso demanda altos custos de energia, pois é preciso manter boa parte da infraestrutura em funcionamento (MOGHADDAM *et al.*, 2015). A grande quantidade de equipamentos ligados aumenta a liberação de calor o que pode elevar bastante a temperatura ambiente do *data center*. Elevadas temperaturas podem levar ao superaquecimento de componentes, aumentar a ocorrência de falhas e reduzir o tempo de vida dos dispositivos presentes no *data center* (NOROUZI-KHANGAH *et al.*, 2016). Para evitar que isso aconteça, os Provedores de Serviço investem em sofisticados mecanismos de climatização que mantêm a temperatura ambiente do *data center* em um nível ótimo de funcionamento. Contudo, tais mecanismos tendem a aumentar ainda mais o consumo e os gastos com energia.

Uma das vertentes de pesquisas que focam em atacar o problema do consumo de energia de um *data center* é o voltado a consolidação de servidores (PROCACCIANTI *et al.*, 2015). Estudos como Fan *et al.* (2007) e Barroso e Hölzle (2007) afirmam que servidores,

enquanto ativos, tendem a ter seus recursos computacionais (processador, memórias e rede) subutilizados. Isso acarreta em um elevado (e desnecessário) consumo de energia. Este consumo pode ser reduzido através da consolidação de servidores. Tal consolidação consiste em alocar a maior quantidade de Máquinas Virtuais em um menor número de servidores. Este tipo de alocação possibilita que as máquinas servidoras que não receberam nenhuma Máquina Virtual sejam desligadas, postas em modo de hibernação ou em qualquer outro modo de baixo consumo de energia.

Embora a consolidação de servidores proporcione economia de energia em um *data center*, ela também tende a aumentar o número de ocorrências de violações de SLA, pois quanto maior o nível de consolidação, ou seja, quanto mais máquinas virtuais alocadas em um único servidor, maior será a concorrência pelos recursos computacionais do equipamento físico. Essa condição de disputa pode prejudicar a elasticidade, por consequência, a Qualidade do Serviço ofertado e o desempenho das aplicações hospedadas pelas Máquinas Virtuais. Assim, a consolidação de servidores precisa ser feita de maneira a equilibrar esses dois objetivos conflitantes: minimizar a energia e maximizar o desempenho.

1.1 Justificativa

Conforme já destacado anteriormente, o consumo de energia em um *data center* é um problema pertinente. Para fins comparativos, o consumo de energia em único *data center* é comparável ao consumo de 25000 domicílios (de Matos *et al.*, 2015 apud BELOGLAZOV; BUYYA, 2012a). Isso equivale a quase 4% do número de domicílios particulares permanentes da cidade de Fortaleza capital do estado do Ceará, segundo o (IBGE, 2010). Segundo Danilak (2017), os *data centers* globais usaram aproximadamente 416 terawatts de energia. Esse valor representa 3% do total da eletricidade utilizada no planeta em 2017. O mesmo autor informa que esse valor é quase 40% a mais do consumo energético de todo o Reino Unido. Tais estudos comprovam a relevância de pesquisas realizadas para reduzir o consumo de energia de um *data center* e, por consequência, o impacto ambiental negativo que ele ocasiona.

O alto consumo de energia de um *data center* produz um efeito colateral importante para o meio ambiente: a alta emissão de gás carbônico (CO_2) na atmosfera e uma grande contribuição ao Efeito Estufa. Segundo Clark (2011) cada busca realizada no Google gera cerca de 0,2g de CO_2 e Vaughan (2015) afirma que assistir 10 minutos de vídeo no YouTube gera o equivalente a 1g de CO_2 . Ainda segundo Vaughan (2015), em 2013, o Google produziu

1.766.014 toneladas de CO_2 , na maior parte gasto pelos seus *data centers*. O mesmo autor também relata que os *data centers* são responsáveis por 2% das emissões de gases na atmosfera terrestre, valor equivalente ao consumido pelas companhias aéreas.

Por isso, é necessário a criação de Políticas de Alocação e de Seleção de Máquinas Virtuais que objetivam minimizar o consumo de energia através da consolidação de servidores. A Política de Seleção será encarregada de selecionar uma Máquina Virtual para ser retirada do servidor e a Política de Alocação escolherá uma nova Máquina Física para abrigar essa Máquina Virtual. Também é importante que, ao proporcionar economia de energia, estas políticas não comprometam o desempenho das aplicações executadas no *data center*, mantendo o número de violações de SLA em um nível controlado. Este trabalho focará em propor uma nova Política de Seleção de Máquinas Virtuais, nomeada AHPSelection, que atuará em conjunto com Políticas de Alocação já propostas pela literatura, já que, como destacado anteriormente, é necessário uma Política de Seleção e uma de Alocação para a realização da migração de Máquinas Virtuais.

1.2 Objetivos

Propor e analisar uma nova Política de Seleção de Máquinas Virtuais em *data centers* afim de reduzir o consumo de energia sem prejudicar o desempenho das aplicações.

1.2.1 Objetivos específicos

- Propor uma nova Política de Seleção de Máquinas Virtuais visando reduzir o consumo de energia do *data center*
- Integrar a Política de Seleção proposta com Políticas de Alocação já existentes.
- Analisar o comportamento da Política no simulador *CloudSim*
- Analisar os resultados do comportamento do algoritmo comparando com outras Políticas de Seleção já existentes

1.3 Organização do Trabalho

O presente trabalho está organizado da seguinte forma: No Capítulo 2, é apresentada a Fundamentação Teórica onde são definidos os conceitos de Computação em Nuvem e Virtualização e sua ligação com as Políticas de Seleção e Alocação. Além disso, é apresentado a técnica AHP utilizada nesse trabalho. No Capítulo 3 são apresentados alguns trabalhos já produzidos

relacionados a Política de Seleção de Máquinas Virtuais. No Capítulo 4 é apresentado a proposta desse trabalho denominada AHPSelection e no Capítulo 5 é definido como será o ambiente e cenários para realização dos testes de validação da política. No Capítulo 6 é analisado os resultados dos testes e no Capítulo 7 é feito uma conclusão sobre o trabalho e desafios para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

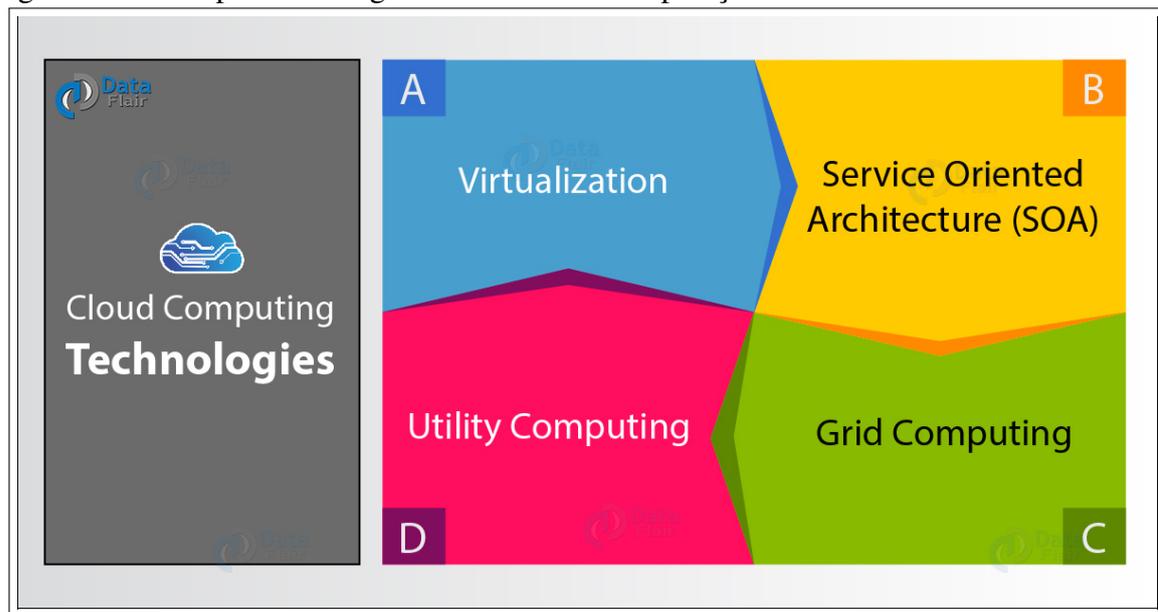
Este capítulo apresenta os fundamentos necessários para o entendimento da Política de Seleção de Máquinas Virtuais apresentada neste trabalho. Inicialmente, serão abordados os conceitos de Computação em Nuvem, Computação em Nuvem Verde e Virtualização. Posteriormente, serão apresentadas os conceitos relacionados a Políticas de Alocação e Políticas de Seleção. Finalmente, será exposta, de forma aprofundada, a técnica base para a Política AHPSelection: a AHP (*Analytic Hierarchy Process*).

2.1 Computação em Nuvem

Segundo Mell e Grance (2011), o modelo de Computação em Nuvem, pode ser definido como um modelo que permite o compartilhamento de um conjunto de recursos de computação que pode ser configurado (por exemplo, redes, armazenamento, servidores, serviços e entre outros) e que pode ser facilmente alocado e liberado pelo usuário sem que haja uma preocupação no gerenciamento ou interação com o Provedor de serviço. Assim, se um usuário desejar, por exemplo, armazenar arquivos, persistir informações ou, até mesmo, executar aplicações de forma remota, através da Internet, alugando temporariamente máquinas com um alto nível de desempenho, disponibilidade e confiabilidade, a melhor solução é aderir a Computação em Nuvem. A Figura 1 mostra as principais tecnologias que serviram como alicerces para a Computação em Nuvem: Computação em Grade (*Grid Computing*), Arquitetura Orientada a Serviços (SOA), *Utility Computing* e a Virtualização.

A Computação em Grade é um modelo de computação distribuída onde junta-se várias máquinas (comumente, com diferentes configurações de hardware e software) interconectadas, através de uma rede de computadores, que compartilham seus recursos e permitem a combinação de poder computacional de todos os membros da *Grid* na resolução de um problema complexo, por exemplo. Imagine um cientista que deseja processar os incontáveis *terabytes* de dados de sua pesquisa. Uma opção seria ele ou a instituição onde trabalha adquirir um computador superpotente (e também bastante caro) para que tais dados possam ser analisados em tempo hábil. Outra opção seria utilizar uma arquitetura em *Grid*, onde várias máquinas comuns (e bem mais baratas) poderiam ser organizadas em rede e cada uma delas processar partes dos dados. Adotar esse tipo de sistema reduz custos, pois equipamentos usados para montar a *Grid*, quando já não estão disponíveis na instituição, são bem mais baratos de adquirir

Figura 1 – Principais tecnologias utilizadas na Computação em Nuvem



Fonte: Data Flair (2019).

que um supercomputador, por exemplo.

A Computação em Nuvem utiliza o mesmo conceito da Computação em Grade, pois ambas compartilham os mesmos objetivos de redução de custos, aumento de flexibilidade e acessibilidade de recursos (VAQUERO *et al.*, 2009). Também, de acordo com o autor, a Computação em Grade trouxe para a Computação em Nuvem a agregação da heterogeneidade dos recursos de hardware e software. A Computação em Grade oculta a heterogeneidade dos recursos, pois sua essência é transparecer um sistema único. A Computação em Nuvem herdou essa característica e adicionou a ela, a possibilidade de virtualizar os recursos de hardware também. Além disso, é possível também a utilização de serviços de terceiros, bem como fornecer os seus serviços a outras aplicações.

Outro alicerce para a Computação em Nuvem foi o *Service-Oriented Architecture* (SOA). SOA é um estilo arquitetural que permite a criação de aplicativos de software baseado em serviços, na qual o serviço é uma implementação de uma funcionalidade de negócios bem definida (MAHMOUD, 2005). Adotar a arquitetura SOA implica em pensar em termos de serviços e aplicações baseadas em serviços, na qual, os recursos de TI são usados para apoiar o processo estratégico do negócio de uma organização (MARZULLO, 2009).

A Computação em Nuvem pode ser vista como uma extensão da SOA, pois integra e fornece aplicações, principalmente para o meio corporativo, como serviços discretos, ou seja, serviços independentes. De acordo com Pontes (2010) a aplicação e a infraestrutura física, assim como os aplicativos em SOA, devem ser descobríveis, gerenciáveis e governáveis. O mesmo

afirma que esses padrões somariam todo o ciclo de vida de uma solução em nuvem. Assim, SOA contribuiu para a Computação em Nuvem porque aborda uma estratégia baseada em serviço, na qual ambas são caracterizadas.

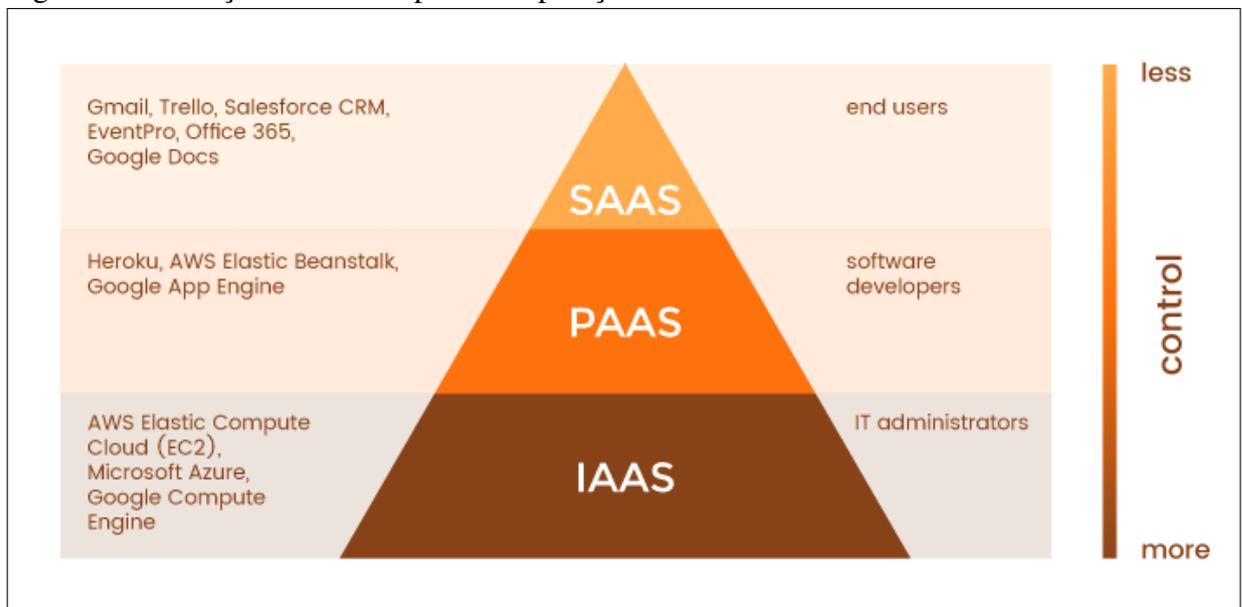
A grande vantagem deste modelo em Nuvem está na forma como o servidor físico é alugado. Para cada recurso alocado ou serviço utilizado pelo Cliente é cobrado um valor, entretanto ele só será tarifado por aquilo que foi consumido. Esta característica é herança da *Utility Computing*. Nela, os recursos computacionais (software e hardware) são oferecidos em forma de serviço, assim como acontece com serviços de telefonia e de distribuição de energia elétrica, água e esgoto, por exemplo. O termo *utility* é usado justamente para fazer analogia com outros serviços, como a energia elétrica, que busca atender a necessidade do cliente e com base no uso que o mesmo faz do serviço, cobrá-lo. Isso é da mesma forma que acontece com a Computação em Nuvem.

Como já citado anteriormente, a grande maioria dos recursos de hardware e software da Nuvem são disponibilizadas como serviço. Embora atualmente existam outros tipos de serviço como, por exemplo, *Database as a Service* (DbaaS), *Network as a Service* (NaaS) e *Container as a Service* (CaaS), os três principais tipos de serviços oferecidos pela Nuvem e mais aceitos pela comunidade acadêmica estão representados na Figura 2 e são:

- **Software como Serviço (SaaS):** Esse é o método de entrega de aplicativos na Internet, onde os Provedores hospedam, provisionam recursos para a execução e gerenciam a execução desses aplicativos. Aos usuários bastam utilizar as aplicações de forma remota através dos seus navegadores ou de um algum programa Cliente oferecido pelo provedor. De acordo com a Figura 2, o *Google Docs* e o *Gmail* são exemplos desse tipo de serviço.
- **Plataforma como Serviço (PaaS):** Neste tipo de serviço, a provedora entrega uma plataforma completa de desenvolvimento de software (além da infraestrutura para executar as aplicações, IDEs de desenvolvimento, sistemas de gerenciamento de banco de dados, sistemas de controle de versão e entre outros) aos clientes. Tais plataformas são usadas pelos desenvolvedores para criar e hospedar aplicações WEB na Nuvem. A vantagem desse tipo de serviço é a possibilidade de desenvolver sem se preocupar com a configuração ou o gerenciamento da infraestrutura física e lógica onde a aplicação será executada. De acordo com a Figura 2, o *Google App Engine* e o *Heroku* são exemplos desse tipo de serviço.

- **Infraestrutura como Serviço (IaaS):** Neste tipo de serviço, o Provedor entrega seus recursos computacionais de infraestrutura como servidores, espaço de armazenamento e capacidade de rede para os clientes. Assim, o cliente pode utilizar estes recursos para, por exemplo, salvar arquivos e dados ou, até mesmo, criar instâncias virtuais com configurações próprias e de acordo com suas necessidades. De acordo com a Figura 2, o *AWS Elastic Compute Cloud* e o *Microsoft Azure* são exemplos desse tipo de serviço.

Figura 2 – Serviços oferecidos pela Computação em Nuvem



Fonte: Gleb B. (2017).

Esses são os principais serviços oferecidos pela Computação em Nuvem. Este trabalho se concentrará somente no cenário de IaaS, onde o Cliente submete uma aplicação para ser executada em uma Máquina Virtual com a configuração desejada por ele. Essa Máquina Virtual está relacionada ao conceito de Virtualização.

2.2 Virtualização

A Virtualização consiste na emulação de ambientes isolados, ou seja, de ambientes de execução independentes, que podem executar diferentes sistemas operacionais, mas que rodam no mesmo equipamento físico, aproveitando assim ao máximo a capacidade do hardware da máquina (COELHO *et al.*, 2008). A Virtualização depende de software para simular a funcionalidade do hardware. Com a Virtualização, pode-se particionar os recursos das máquinas

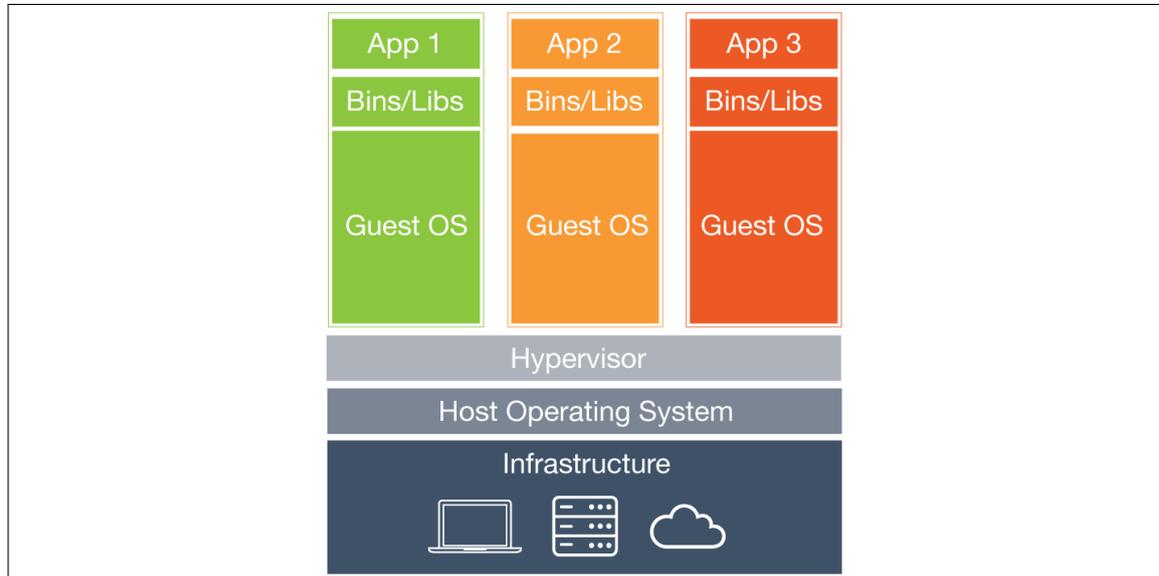
físicas, ou seja, executar diversas aplicações em uma mesma máquina, dividindo seu poder computacional. A Computação em Nuvem se beneficia também de características como: o *sandboxing*, onde cada Máquina Virtual possui seu isolamento, ou seja, é separada uma das outras, de forma que caso ocorram erros ou problemas de segurança, esses problemas não se espalharão e não afetarão o desempenho das demais instâncias virtuais; o *pool* de recursos (falsa impressão de recursos ilimitados); a elasticidade (a flexibilidade em redimensionar os recursos de uma instância virtual em tempo de execução) e a independência de hardware (as instâncias virtuais funcionam corretamente independentemente da arquitetura e das especificações do hardware que as estejam executando).

Na Virtualização há dois tipos de máquinas: a Física e a Virtual. A Máquina Física compreende uma máquina real, ou seja, aquela que está fisicamente presente em um *data center*. Tais Máquinas Físicas podem ser desde equipamentos de rede a servidores terminais. A Máquina Virtual, por sua vez, não existe fisicamente, trata-se de uma instância virtualizada da Máquina Física aonde ela está alocada. Essa monografia lidará apenas com Máquinas Físicas e Virtuais relacionadas a servidores.

Para a Virtualização, é preciso saber as diferentes camadas que ajudam esse processo a funcionar. O sistema Hospedeiro (SO *Host*) é o sistema operacional executado na máquina física. O sistema Hóspede (SO *Guest*), é o sistema operacional virtualizado, presente na máquina virtual, que deve ser executado pelo sistema hospedeiro. Para intermediar esses dois, há o *Hypervisor*. O *Hypervisor* trata-se de uma espécie de plataforma implementada no hospedeiro que recebe os sistemas a serem virtualizados, controlando os seus recursos, gerenciando a execução das Máquinas Virtuais e mantendo o isolamento entre eles.

A Figura 3 mostra graficamente uma das formas de organizar os principais componentes relacionados a Virtualização:

Figura 3 – Estrutura tradicional de virtualização



Fonte: Aouar (2018).

A principal tecnologia que visa a eficiência energética em um *data center* é a Virtualização (PANDYA, 2014). A Virtualização permite executar mais aplicações em menos servidores. Logo, menos servidores ligados, se desconsiderarmos superaquecimentos ou zonas de calor, menor será o consumo de energia. Mais aplicações por servidor, permite desligar servidores sem carga de trabalho ou ao menos colocá-los em modo de baixo consumo de energia. Essa prática adotada na Computação em Nuvem está relacionada ao conceito de Computação em Nuvem Verde.

2.3 Computação em Nuvem Verde

As vantagens apresentadas na seção anterior atraíram a atenção e o investimento de todos (Clientes e Provedores) para soluções voltadas para a Nuvem. Esse aumento de interesse levou a uma proliferação da Computação em Nuvem e resultou no estabelecimento de *data centers* em larga escala em todo o mundo.

Segundo Hussein *et al.* (2014), os *data centers* possuem elevados custos operacionais e consomem enormes quantidades de energia que contribuem nas emissões de carbono para o meio ambiente. O dióxido de carbono e outros poluentes do aquecimento global estão se acumulando na atmosfera, fazendo com que o planeta se aqueça. Também, de acordo com o

trabalho, a indústria da Tecnologia da Informação é altamente responsável por uma grande parte do crescimento mundial do consumo de energia causado por milhares de *data centers* e sistemas de computação em todo o mundo.

A Computação em Nuvem Verde visa montar uma infraestrutura que combine flexibilidade, Qualidade de Serviço e utilização reduzida de energia (WESTPHALL; VILLARREAL, 2013). Esse novo paradigma quer ofertar os mesmos benefícios que a Computação em Nuvem tradicional fornece, como economia e elasticidade, mas aplicando técnicas que minimizem o impacto ambiental como, por exemplo, uso de fontes de energia renováveis, virtualização de servidores e readequação da infraestrutura elétrica de refrigeração. Porém, é importante que o uso destas e outras técnicas também garantam que o serviço prestado pela Nuvem não perca qualidade. Logo, a Computação em Nuvem Verde é definida como um conjunto de metodologias que reduzem o impacto de projetar, fabricar, usar e descartar de maneira eficiente sistemas de computação, rede e comunicações (Hussein *et al.*, 2014).

Existem várias técnicas e ferramentas que podem ser aplicadas e associadas a Computação Verde, por exemplo, soluções inteligentes para posicionamento geográfico de *data centers* (Hussein *et al.*, 2014), reciclagem de equipamentos (PENSAMENTO VERDE, 2013) e alocação de *data centers* onde a energia é mais barata (WESTPHALL; VILLARREAL, 2013). Para se ter uma Computação em Nuvem Verde é preciso atacar os diversos problemas que existem. Um dos problemas mais importantes na Computação em Nuvem é o consumo de energia (Qiu *et al.*, 2018 apud Niu *et al.*, 2012).

Uma das formas de minimizar o problema do alto consumo energético e contribuir com o *data center* com consciência energética é através da Virtualização. Considere, como exemplo, um cenário onde um *data center* precisa executar três aplicações distintas. Em um ambiente não virtualizado, cada aplicação seria entregue a um servidor físico exclusivo, mesmo que elas consumissem pouquíssimos recursos computacionais durante todo seu ciclo de vida. Já em um ambiente virtualizado, todas as três aplicações poderiam ser alocadas em uma única máquina física, permitindo que os demais servidores pudessem ser desligados ou postos em modo de economia de energia.

Contudo, os recursos computacionais de uma máquina física são finitos. Dessa maneira, não é possível alocar todas as aplicações em um único servidor. À medida que se incrementa o número de aplicações em um único servidor, também se aumenta a disputa pelos recursos físicos e isso tende a levar uma degradação no desempenho do serviço prestado. Assim

se faz necessário selecionar e distribuir as aplicações (hospedadas em Máquinas Virtuais) entre os diversos servidores que compõe o *data center* da melhor forma possível. Essas atividades são papéis das Políticas de Alocação e de Seleção de Máquinas Virtuais.

2.4 Políticas de Alocação e Seleção

Um *data center* em Nuvem usa a tecnologia de Virtualização para alocar várias Máquinas Virtuais em servidores físicos e pode, eventualmente, migrar Máquinas Virtuais (ou seja, mover uma Máquina Virtual sem desligá-la) entre servidores para equilibrar cargas de trabalho (processador, memória, armazenamento e largura de banda de rede) entre eles (MOGHADDAM *et al.*, 2018 apud CLARK *et al.*, 2005).

Para alocar Máquinas Virtuais é preciso escolher Máquinas Virtuais e Físicas ideais respeitando algum critério como, por exemplo, reduzir o consumo de energia, ou assegurar o desempenho e minimizar a violação de SLA, ou até mesmo, os dois. A tarefa de escolher uma máquina física ideal para receber uma Máquina Virtual é delegada a Política de Alocação. Mas, anteriormente, é preciso ter uma Máquina Virtual ideal a ser migrada. Essa tarefa é dada a política de seleção que é o foco deste trabalho.

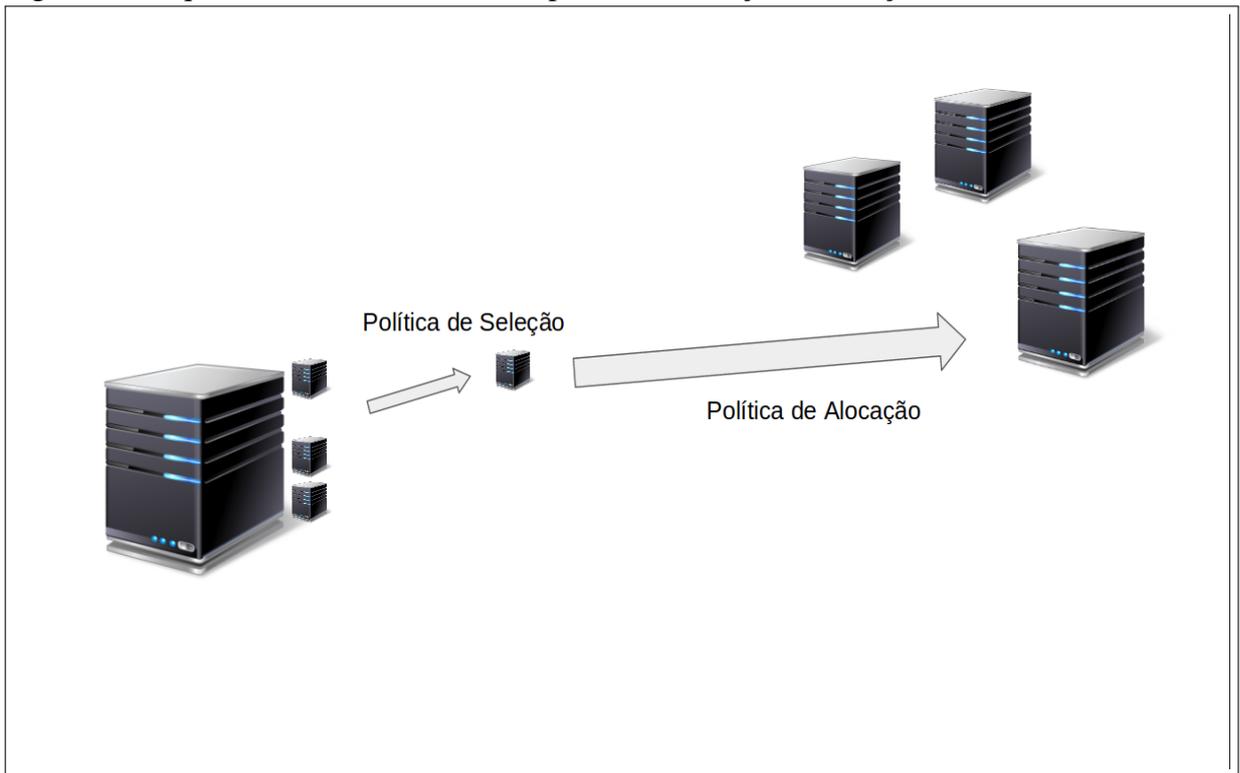
Para ilustrar o funcionamento dessas políticas, observe o seguinte cenário. Suponham que existam várias Máquinas Físicas em um *data center*. Cada uma delas, abrigam diversas Máquinas Virtuais. Mas, é nótório afirmar e é sabido que as Máquinas Físicas possuem limitações de recursos e podem em algum momento perderem a capacidade de abrigar mais Máquinas Virtuais. Se isso acontece, é preciso redistribuir essas Máquinas Virtuais para que os servidores não fiquem sobrecarregados.

Suponha um cenário em que uma Máquina Física está sobrecarregada, ou seja, chegou ao limite de sua capacidade e recursos. Para isso, é preciso fazer com que uma ou mais Máquinas Virtuais sejam migradas, ou seja, direcionadas de uma máquina para outra, já que a máquina está sobrecarregada.

Para escolher a Máquina Virtual que será migrada entre as Máquinas Virtuais existentes no servidor é utilizada a Política de Seleção. Selecionada a Máquina Virtual é preciso definir seu destino, ou seja, a nova Máquina Física que irá abrigá-la. Para este fim, é utilizada a Política de Alocação. A Figura 4 mostra de maneira esquematizada o processo de migração utilizando a política de seleção e, logo após, a de alocação.

Este trabalho terá como foco propor uma nova Política de Seleção que utilizará

Figura 4 – Esquema do funcionamento da política de seleção e alocação



Fonte: Próprio autor

como ferramenta o método *Analytic Hierarchy Process* (AHP). Na próxima seção será detalhado melhor sobre a disciplina em que a AHP é contida e a relação da AHP com o Problema de Programação Linear.

2.5 Pesquisa Operacional

A Pesquisa Operacional é uma disciplina que auxilia na tomada de decisão de problemas complexos com uso de modelos, seja quantitativos ou qualitativos (MONKS, 2016). Neste trabalho, como se trata de uma Política de Seleção, é preciso escolher uma Máquina Virtual entre vários outras para a migração. Essa escolha se baseia em uma tomada de decisão. "*O problema de tomada de decisão leva em conta variáveis e as condições que "prendem" estas variáveis, às quais denominaremos restrições*" (FILHO; PEREIRA, 1980). Na Pesquisa Operacional existem diversos problemas e entre eles, o Problema de Programação Linear (PPL).

2.5.1 Problema de Programação Linear

Problemas de Programação Linear (PPL) são problemas que consistem em uma função linear a ser maximizada ou minimizada sendo sujeita a restrições na forma de equação ou inequação linear (KAHLIG, 2019). Para resolver um PPL é preciso maximizar ou minimizar uma função linear com variáveis de decisão contínuas que são chamadas de função objetivo, tendo um conjunto de restrições, formado por um sistema linear de igualdades ou desigualdades. Na próxima subseção é apresentado a formulação do modelo geral para representar um PPL.

2.5.2 Representação de um Problema de Programação Linear

Um Problema de Programação Linear é formulado de modo a identificar um conjunto de variáveis não-negativas que minimizem ou maximizem uma função objetivo linear sujeita a um conjunto de restrições lineares. Pode ser definido sob a seguinte forma:

Função objetivo

$$Z = \max \sum_{j=1}^n c_j x_j \quad (2.1)$$

ou

$$Z = \min \sum_{j=1}^n c_j x_j \quad (2.2)$$

Sujeito a

$$\sum_{j=1}^n a_{ij} x_j \leq b_i \quad i = 1, 2, \dots, m \quad (2.3)$$

$$x_j \geq 0 \quad j = 1, 2, \dots, n \quad (2.4)$$

2.5.3 Técnicas para resolução de PPL

O problema de seleção de Máquinas Virtuais se encaixa como Problema de Programação Linear, já que pode ser modelado para uma função objetiva de maximização dos critérios.

Algumas técnicas são utilizadas para resolver Problemas de Programação Linear tais como os métodos analíticos e o método Simplex. A AHP é uma técnica para resolução de Problemas de Programação Linear. Na modelagem para a AHP, existe um conjunto que contém todas as máquinas virtuais e um conjunto que possui todos os critérios e subcritérios. A função objetivo para o AHP seria maximizar o total das prioridades dos critérios das Máquinas Virtuais. Na próxima seção, é detalhado melhor sobre a técnica AHP.

2.6 AHP - *Analytic Hierarchy Process*

Durante a vida, muitas vezes, as pessoas são postas em situações de tomadas de decisões. Entende-se como tomada de decisão o processo de se resolver um problema escolhendo, dentre as alternativas de escolha, as melhores soluções possíveis para aquele problema (PORTO; BANDEIRA, 2006). Toda decisão é baseada em julgamentos ou critérios. Estes julgamentos são subjetivos e podem conduzir a vida do julgador a diferentes caminhos que podem fazê-la progredir ou retroceder. Por conta disso, é de suma importância realizar um bom julgamento nestes momentos.

Para tomar boas decisões, segundo Saaty (2008), deve-se, além de se coletar dados sobre o problema em análise, organizar todas as informações obtidas de uma forma estruturada, visando facilitar as tomadas de decisão. Para isso, alguns métodos são utilizados para auxiliar neste procedimento. O método AHP (*Analytic Hierarchy Process*) é um deles. O AHP é um método de tomada de decisão e de classificação usando hierarquia sob múltiplos critérios (NEVES; CAMANHO, 2015). O método AHP compreende os seguintes passos:

1. Definir o problema e o tipo de conhecimento
2. Definir os critérios e subcritérios
3. Definir as alternativas
4. Definir graus de importâncias e prioridades entre critérios, subcritérios e alternativas
5. Compilar tudo e definir prioridades gerais das alternativas

No método AHP, inicialmente, é necessário contextualizar o problema a ser atacado e que tipo de conhecimento se deseja alcançar para solucioná-lo, ou seja, o objetivo final da tomada de decisão. De uma forma geral, esta é a parte impessoal do método, pois a descrição do problema e o tipo de conhecimento em foco são comuns a mais de um indivíduo.

Após estas definições iniciais, inicia-se a parte pessoal do método, onde o indivíduo irá elencar e ponderar uma série de opiniões e visões próprias sobre o problema e o tipo de conhecimento levantado. O primeiro passo dessa etapa é o estabelecimento dos critérios relevantes ao julgador durante a tomada de decisão. Eventualmente, pode-se definir um conjunto de subcritérios relacionados aos critérios iniciais ou, até mesmo, a outros subcritérios.

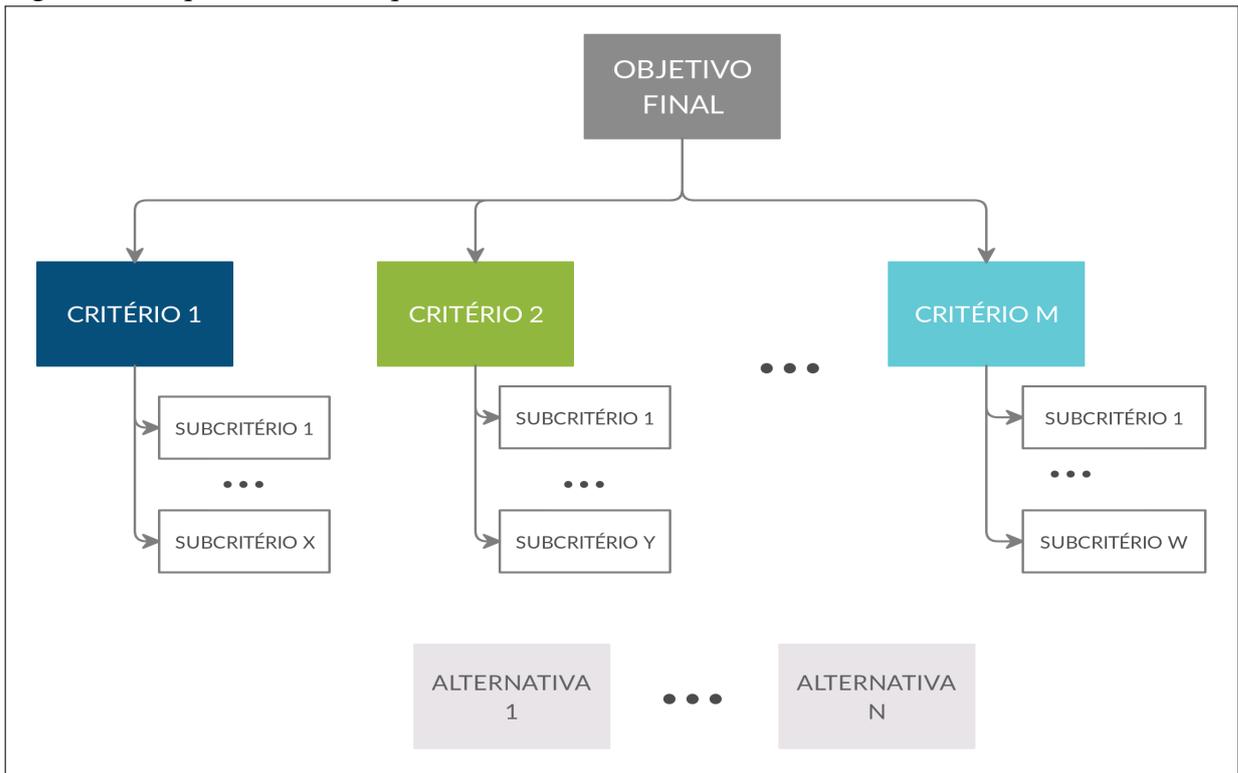
Um processo de tomada de decisão é composto por opções de escolha ou, de acordo com a terminologia do método AHP, as alternativas no modelo. As alternativas são as opções de escolha disponíveis ao julgador onde uma ou várias delas serão selecionadas com base nos critérios e, se presentes, no subcritérios definidos na etapa anterior.

O próximo passo é construir uma hierarquia para definir de forma estruturada o tipo de conhecimento, os critérios e as alternativas levantadas. No primeiro nível é apresentado o objetivo final, ou seja, o tipo de conhecimento que se deseja adquirir. No nível abaixo estão presentes os critérios e, eventualmente, subcritérios e no último nível as opções de escolha. Esta visão hierárquica visa facilitar a análise do julgador acerca da completude de sua reflexão sobre aquilo que é necessário para tomar a decisão, bem como fazer algumas correções de julgamento antes de partir para as etapas mais complexas da técnica. O esquema da hierarquia é representado na Figura 5

Uma vez montada a hierarquia, é preciso ponderar cada critério disponível. Julgar os critérios e atribuir pesos a cada um deles é uma tarefa bastante subjetiva e é impossível sem uma ligação direta com uma escala. Cada escala relaciona um valor ou peso a um possível julgamento. A Tabela 1 exemplifica uma dessas escalas. Tal escala foi proposta por Saaty (2008), criador do AHP, que a define como a escala fundamental de números absolutos. Através dela é possível mensurar e comparar diferentes níveis de importância e assim extrair a dominância e a prioridade de cada critério ou alternativa uns com os outros.

No método AHP, são construídas matrizes que guardam esses valores quantificados baseados na escala de (SAATY, 2008). Para o método, cada elemento a_{ij} representa o valor guardado na linha i e coluna j da matriz, que é utilizado para quantificar a importância entre os critérios i e j , alternativas i e j (com base em um critério específico) ou, em uma etapa mais avançada, de uma alternativa i para um critério j . É preciso que esses valores estejam consistentes, ou seja, um valor dado a um campo precisa ter o mesmo significado e não ser contrariado em outros campos. Afim de assegurar a consistência assume-se que $a_{ii} = 1$ para todo i , e $a_{ji} = 1 / a_{ij}$. Se o julgamento for consistente, a matriz satisfará o critério de consistência

Figura 5 – Esquema de hierarquia AHP



Fonte: Próprio autor

dado por 2.5:

$$a_{ij} * a_{jk} = a_{ik} \quad (2.5)$$

A matriz precisa ter consistência não só nos dados, mas nos julgamentos feitos a cada critério, subcritério ou alternativa. Para verificar a consistência, Saaty (1984) elaborou duas métricas, o Índice de Consistência (IC) e a Razão de Consistência (RC), que auxiliam, de maneira matemática, a determinar a consistência ou não dos julgamentos e dos valores da matriz. Nas próxima subseção será detalhado como obter essas métricas.

2.6.1 Índice de Consistência (IC) e Razão de Consistência (RC)

Saaty (1984) sugere que, para verificar a consistência dos julgamentos, seja calculada o Índice de Consistência (IC) e a Razão de Consistência da matriz.

O Índice de Consistência (IC) é calculado pela Fórmula 2.6.

$$IC = \frac{\lambda_{max} - n}{n - 1} \quad (2.6)$$

Onde:

Tabela 1 – Escala fundamental de números absolutos

Intensidade de Importância	Definição	Explicação
1	Igual importância	Dois atividades contribuem igualmente para o objetivo
2	Fraco ou leve	
3	Importância moderada	Experiência e julgamento favorecem ligeiramente uma atividade em detrimento de outra
4	Mais moderado	
5	Forte importância	Experiência e julgamento favorecem fortemente uma atividade sobre outra
6	Importância mais forte	
7	Importância muito forte	Uma atividade é favorecida muito fortemente em detrimento de outra; seu domínio é demonstrado na prática
8	Muito, muito forte	
9	Extrema importância	A evidência que favorece uma atividade em detrimento de outra é da mais alta ordem possível de afirmação
Recíprocos de cima	Se a atividade i tiver um dos números acima de zero atribuídos a ela quando comparada com a atividade j, então j tem o valor recíproco quando comparado com i	
1.1–1.9	Se as atividades são muito próximas	Pode ser difícil atribuir o melhor valor, mas quando comparado com outras atividades contrastantes o tamanho dos números pequenos não seria muito perceptível, mas eles ainda podem indicar a importância relativa das atividades.

Fonte: Adaptado de (SAATY, 2008)

λ_{max} = autovalor máximo da matriz

n = Número de alternativas/opções

De acordo com o Saaty (1984), o autovalor mede a consistência do julgamento.

Logo, λ_{max} é calculado como o autovalor máximo.

O Índice de Consistência é utilizado para calcular a Razão de Consistência (RC). O cálculo da RC é dado pela Fórmula 2.7.

$$RC = \frac{IC}{IR} \quad (2.7)$$

Onde:

$IC = \acute{\text{I}}ndice \text{ de } Consist\acute{\text{e}}ncia$

$IR = \acute{\text{I}}ndice \text{ Rand\acute{o}}mico \text{ para } n$

Saaty (1984) definiu uma tabela com os índices randômicos de matrizes de ordem 1 a 15, calculados em laboratórios, conforme está sendo exibido na Figura 6.

Figura 6 – Índice randômico

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.48	1.56	1.57	1.59

Fonte: (SAATY, 1984)

Saaty (1984) fez uma proposta de que se o valor de RC for maior ou igual a 0.1, os julgamentos estão inconsistente. Caso contrário, está consistente.

Feito essa quantificação dos julgamentos, é possível analisar e definir a prioridade de cada um deles, bem como extrair a relação de dominância de um critério em relação a outro. Para se calcular tais prioridades, são aplicadas diversas ferramentas matemáticas e, uma delas, é a média geométrica. A média geométrica é utilizada para encontrar o autovetor, já que esse dá a ordem de prioridade. De acordo com as simulações de BAJWA *et al.* (2008), a média geométrica é o melhor método para se obter as prioridades em uma matriz. Visando facilitar a comparação entre critérios e alternativas, Saaty (2008) sugere que as médias geométricas sejam normalizadas. As próximas sessões explicam de forma resumida esses dois conceitos.

2.6.2 Média Geométrica

A ideia original da média geométrica foi desenvolvida por Crawford e Williams (1985) e consiste, de forma bem resumida, no produto de todos os números do conjunto elevado ao inverso da quantidade de membros desse conjunto. A média geométrica de um vetor v_i é dado pela Fórmula 2.8:

$$v_i = \prod_{j=1}^n a_{ij}^{1/n} \quad (2.8)$$

Onde:

v_i = Média geométrica do vetor de critério i

n = Número de colunas ou elementos do vetor

a_{ij} = O valor do vetor na linha i e coluna j

2.6.3 Normalização

Afim de facilitar as comparações entre os valores encontrados pela média geométrica, é utilizada uma fórmula para a normalização dos mesmos. Normalizar é transformar valores com diferentes escalas de medidas e ajusta-los a uma escala de medida comum, por exemplo, em uma escala entre 0 e 1 (ambos inclusos). A normalização é consequência da média geométrica, pois a utiliza como base para a obtenção de seu resultado. A normalização é obtida através da Fórmula 2.9:

$$p_i = \frac{v_i}{\sum_{j=1}^n v_j} \quad (2.9)$$

Onde:

p_i = Valor normalizado da média geométrica do vetor de critério ou alternativa i

v_i = Média geométrica do vetor de critério ou alternativa i

v_j = Média geométrica do vetor de critério ou alternativa j

n = Número de critérios ou alternativas

Para melhor compreender o AHP, desde sua definição, construção e resultado final, será utilizado um exemplo de tomada de decisão. Na próxima seção será detalhado o problema, sua estruturação na AHP e o resultado final.

2.7 AHP através de um exemplo

Para melhor compreensão e exemplificação da construção e dos resultados obtidos pelo método AHP, Saaty (2008) propõe o seguinte exemplo: Suponha que uma pessoa tenha terminado seu doutorado e está à procura de emprego. Mas, esta fica em dúvida, quanto ao trabalho, se é melhor e mais vantajoso trabalhar em dois tipos de empresa ou ensinar em dois tipos de escolas.

De acordo com o método AHP, o primeiro passo é descrever de uma forma geral o problema a ser atacado, o que já foi feito no parágrafo anterior. O segundo passo é definir o tipo de conhecimento que se deseja alcançar. Dessa forma, de acordo com a descrição do problema, o tipo de conhecimento é encontrar o melhor trabalho para o indivíduo em questão.

O terceiro e o quarto passos são levantar os critérios e as alternativas que guiarão o indivíduo na sua tomada de decisão. Na descrição do problema, não estão definidos os critérios,

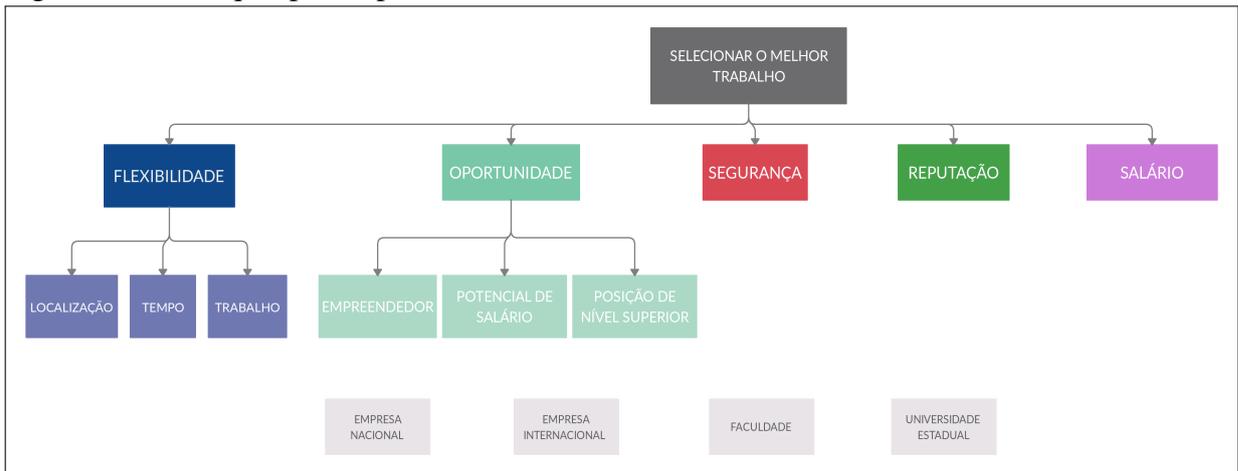
nem os subcritérios, ficando a cargo do indivíduo defini-los. O trabalho de Saaty (2008) sugere alguns desses critérios e subcritérios como, por exemplo, o critério de Flexibilidade que possui como subcritérios a Localização, o Tempo e o Trabalho. Já as alternativas estão presentes na definição do problema e são:

1. Trabalhar em um Companhia Nacional
2. Trabalhar em uma Empresa Internacional
3. Lecionar em uma Faculdade Privada
4. Lecionar em uma Universidade Estadual

De posse de todas estas informações, o próximo passo é construir uma estrutura hierárquica que una tudo aquilo que foi identificado nos passos anteriores. A Figura 7 mostra uma possível hierarquia para o problema do exemplo supracitado. No topo observa-se o objetivo (escolher o melhor emprego) e, conectados a ele, encontram-se os critérios levantados como, por exemplo, o Salário Pago e a Reputação do Local. Também é possível perceber que os critérios podem ser organizados em múltiplas camadas (aqui chamados de subcritérios), desde que os critérios da camada imediatamente inferior estejam de alguma forma ligados a um dos critérios da camada imediatamente superior. Por exemplo, o critério de Flexibilidade possui três subcritérios relacionados (Localização, Tempo e Forma de Trabalho). Tais subcritérios acontecem pois o indivíduo do exemplo, durante sua tomada de decisão, deseja ponderar esses três tipos de Flexibilidade e, obviamente, ele irá priorizar aquele subcritério que mais lhe atrai. Por fim, na parte inferior da hierarquia, descreve-se as alternativas que o indivíduo tem na tomada de decisão: trabalhar em uma empresa nacional ou em uma empresa internacional ou lecionar em uma faculdade privada ou em uma universidade pública. Vale ressaltar que esta estrutura hierárquica é apenas uma das diversas formas de organizar a tomada de decisão do problema. Outro indivíduo, por exemplo, poderia criar uma estrutura diferente para o mesmo problema, com outros critérios e alternativas.

Uma vez elaborada a estrutura hierárquica, o próximo passo é julgar cada um dos critérios, ou seja, ponderar o nível de importância de cada um deles. O ponderamento é feito com base em uma escala previamente definida. Neste exemplo, foi adotado a escala fundamental de números absolutos (Tabela 1). Com base no julgamento feito, monta-se uma matriz onde os critérios são distribuídos entre linhas e colunas e cada elemento representa o nível de importância de um critério em relação ao outro. Por exemplo, de acordo com a Tabela 2, o Salário tem importância 8 em relação a Flexibilidade, ou seja, o Salário é quantificado como muito mais

Figura 7 – Hierarquia para o problema



Fonte: Adaptado de (SAATY, 2008)

importante que a Flexibilidade. Mas, observe que, para manter a consistência, é preciso valorar a Flexibilidade em relação ao Salário como o inverso da relação acima citada. No exemplo, a valoração Flexibilidade/Salário é de 1/8.

Tabela 2 – Matriz de comparação dos critérios

	Flexibilidade	Oportunidade	Segurança	Reputação	Salário
Flexibilidade	1	1/4	1/6	1/4	1/8
Oportunidade	4	1	1/3	3	1/7
Segurança	6	3	1	4	1/2
Reputação	4	1/3	1/4	1	1/7
Salário	8	7	2	7	1

Fonte: Adaptado de (SAATY, 2008)

Uma vez definidos os níveis de importância, o próximo passo é definir as prioridades entre os critérios. Para isso, acrescenta-se uma nova coluna a matriz da Tabela 2, denominada Prioridade. Os valores da Prioridade são obtidos, inicialmente, através do cálculo da média geométrica de cada linha e, posteriormente, visando uma melhor comparação das prioridades, o autor sugere que elas sejam normalizadas.

Considere a linha do critério da Flexibilidade como exemplo. O primeiro passo é calcular a média geométrica da linha em questão utilizando a Fórmula 2.8. Destrinchando o produto da média geométrica, o resultado final será:

$$(a_{11} * a_{12} * a_{13} * a_{14} * a_{15})^{\frac{1}{5}} = (1 * \frac{1}{4} * \frac{1}{6} * \frac{1}{4} * \frac{1}{8})^{\frac{1}{5}} = \sqrt[5]{1 * \frac{1}{4} * \frac{1}{6} * \frac{1}{4} * \frac{1}{8}} \approx 0.2648$$

O mesmo procedimento é feito para as demais linhas (isto é, para cada critério) da Tabela 2.

Feito isso, é necessário calcular o valor normalizado de cada média geométrica utilizando a Fórmula 2.9 . Assim, para se calcular o valor normalizado, primeiramente, soma-se as médias geométricas de todas as linhas obtidas no passo anterior. Por fim, divide-se a média geométrica individual de cada critério por este somatório. O resultado obtido será a prioridade do critério em análise. Tome a Flexibilidade como critério 1. Substituindo os dados na fórmula especificada, obtém-se:

$$p_i = \frac{v_i}{\sum_{j=1}^n v_j}$$

$$p_1 = \frac{v_1}{\sum_{j=1}^5 v_j}$$

$$p_1 = \frac{0,2648}{7,5423}$$

$$p_1 \approx 0,036$$

Assim como o passo anterior, este procedimento se repete para cada um dos critérios.

A Tabela 3 mostra a nova tabela com a coluna Prioridade adicionada.

Tabela 3 – Matriz de comparação dos critérios com a coluna de Prioridade

	Flexibilidade	Oportunidade	Segurança	Reputação	Salário	Prioridade
Flexibilidade	1	1/4	1/6	1/4	1/8	0,036
Oportunidade	4	1	1/3	3	1/7	0,122
Segurança	6	3	1	4	1/2	0,262
Reputação	4	1/3	1/4	1	1/7	0,075
Salário	8	7	2	7	1	0,506

Fonte: Adaptado de (SAATY, 2008)

Se houver subcritérios, o processo é o mesmo: constrói-se uma matriz contendo todos os subcritérios (nas linhas e nas colunas), julga-se o nível de importância entre eles e calcula-se as Prioridades seguindo os passos supracitados. A Tabela 4 exemplifica este procedimento tomando como base os subcritérios vinculados a Flexibilidade.

Tabela 4 – Matriz de comparação dos subcritérios

	Localização	Tempo	Trabalho	Prioridade
Localização	1	1/3	1/6	0,091
Tempo	3	1	1/4	0,218
Trabalho	6	4	1	0,691

Fonte: Adaptado de (SAATY, 2008)

Feito esses passos, agora é preciso julgar cada opção (ou alternativa) com base nos critérios e subcritérios elencados. A Tabela 5 mostra um exemplo disso. Tome como base o critério de Salário Pago. Ao quantificar as alternativas de trabalho com base neste critério, a Companhia Nacional apresenta um melhor desempenho quando comparado a Empresa Internacional. Dessa maneira, foi atribuído peso 4 na relação Companhia Nacional/Empresa Internacional e o inverso ($\frac{1}{4}$) na relação Empresa Internacional/Companhia Nacional. Perceba que também calcula-se, de forma semelhante ao que foi descrito anteriormente, as Prioridades de cada alternativa. Esse processo se repete para cada critério e subcritério levantado.

Tabela 5 – Matriz de comparação das alternativas com base no critério de Potencial de Salário

	Empresa Nacional	Empresa Internacional	Faculdade	Universidade Estadual	Prioridade
Empresa Nacional	1	4	3	6	0,555
Empresa Internacional	1/4	1	3	5	0,258
Faculdade	1/3	1/3	1	2	0,124
Universidade Estadual	1/6	1/5	1/2	1	0,064

Fonte: Adaptado de (SAATY, 2008)

Dado que a matriz de comparação entre os critérios (Tabela 3), a matriz de comparação dos subcritérios (como a Tabela 4) e todas as matrizes de comparação das alternativas com o respectivo critério (como na Tabela 5), é preciso calcular a Prioridade Global, ou seja, unir todas essas informações e obter uma prioridade final. Na Figura 8 é representado a matriz final com suas prioridades finais. Para que se possa juntar todos esses dados a fim de produzir um resultado final, é preciso aplicar alguns passos.

Figura 8 – Matriz de comparação final

Critério	Flexibilidade			Oportunidade			Segurança	Reputação	Salário	PRIORIDADE FINAL
	0.036			0.122						
Subcritério	Localização	Tempo	Trabalho	Empreendedor	Potencial de salário	Posição de nível superior				
	0.091	0.218	0.691	0.105	0.637	0.258				
Pesos globais (Critério x Subcritério)	0.003	0.008	0.025	0.013	0.078	0.032	0.262	0.075	0.506	
Empresa Nacional	0.295	0.084	0.062	0.090	0.555	0.591	0.225	0.064	0.124	0.193
Empresa Internacional	0.496	0.055	0.115	0.061	0.258	0.274	0.054	0.101	0.547	0.333
Faculdade	0.131	0.285	0.249	0.239	0.124	0.083	0.095	0.247	0.289	0.214
Universidade Estadual	0.078	0.576	0.574	0.610	0.064	0.052	0.626	0.588	0.039	0.262

Fonte: Adaptado de (SAATY, 2008)

Primeiramente, é preciso obter os pesos globais (finais) dos critérios e subcritérios.

Para os subcritérios, basta multiplicar cada prioridade do subcritério com a prioridade do critério acima dele. Por exemplo, para se obter o peso global do subcritério Localização (0,003), foi preciso multiplicar a prioridade de Localização (0,091) e a prioridade do critério relacionado a ela, que no caso, é a Flexibilidade (0,036). Se o critério não houver subcritério, basta repetir o valor do critério (como, por exemplo, é o caso do critério Salário). Esse procedimento é feito para todos os critérios e subcritérios. No final, gera-se um vetor com esses resultados que na Figura 8 é nomeado de Pesos Globais.

Observe que os pesos globais dos critérios e subcritérios já estão em seu estado final. Mas, ainda faltam os pesos globais das alternativas. Para isso, gera-se uma matriz onde suas colunas são as prioridades das alternativas de acordo com o critério ou subcritério da coluna e as linhas são as próprias alternativas. A matriz gerada para o exemplo é representada abaixo:

$$\begin{pmatrix} 0,295 & 0,084 & 0,062 & 0,090 & 0,555 & 0,591 & 0,225 & 0,064 & 0,124 \\ 0,496 & 0,055 & 0,115 & 0,061 & 0,258 & 0,274 & 0,054 & 0,101 & 0,547 \\ 0,131 & 0,285 & 0,249 & 0,239 & 0,124 & 0,083 & 0,095 & 0,247 & 0,289 \\ 0,078 & 0,576 & 0,574 & 0,610 & 0,064 & 0,052 & 0,626 & 0,588 & 0,039 \end{pmatrix}.$$

Para se obter a prioridade global das alternativas, basta multiplicar essa matriz gerada com o vetor de pesos globais gerados anteriormente. A representação dessa multiplicação para este problema é descrito abaixo:

$$\begin{pmatrix} 0,295 & 0,084 & 0,062 & 0,090 & 0,555 & 0,591 & 0,225 & 0,064 & 0,124 \\ 0,496 & 0,055 & 0,115 & 0,061 & 0,258 & 0,274 & 0,054 & 0,101 & 0,547 \\ 0,131 & 0,285 & 0,249 & 0,239 & 0,124 & 0,083 & 0,095 & 0,247 & 0,289 \\ 0,078 & 0,576 & 0,574 & 0,610 & 0,064 & 0,052 & 0,626 & 0,588 & 0,039 \end{pmatrix} * \begin{pmatrix} 0,003 \\ 0,008 \\ 0,025 \\ 0,013 \\ 0,078 \\ 0,032 \\ 0,262 \\ 0,075 \\ 0,506 \end{pmatrix}.$$

O resultado dessa multiplicação é um vetor, na qual é chamado de vetor de prioridade global como é mostrado na coluna Prioridade Global na Figura 8 e também abaixo.

$$\begin{pmatrix} 0,193 \\ 0,333 \\ 0,214 \\ 0,262 \end{pmatrix}.$$

A alternativa com maior valor é definida como a alternativa com maior prioridade. Para o exemplo, de acordo com a Figura 8, a Empresa Internacional tem maior valor (0,333), portanto esta é a melhor escolha para a tomada de decisão desse problema. Logo, nesse caso, o melhor trabalho para o indivíduo é trabalhar na Empresa Internacional.

3 TRABALHOS RELACIONADOS

Nesta seção, serão apresentados os trabalhos relacionados a Política de Seleção de Máquinas Virtuais AHPSelection. Tais trabalhos foram coletados através de bibliotecas científicas digitais confiáveis e já amplamente utilizadas na academia (em especial, a IEEE Xplore ¹) utilizando como critérios de aceitação: 1) A classificação do local onde o trabalho foi publicado (de acordo com o *Qualis* ²); 2) O ano da publicação e 3) Se os trabalhos utilizavam o simulador *CloudSim* como ferramenta para a validação de suas propostas. Assim, optou-se por trabalhos que apresentassem propostas de políticas para o problema de seleção de Máquinas Virtuais com classificação de *Qualis* igual ou superior a B1. Da mesma forma, foi priorizado trabalhos com data de publicação igual ou superior ao ano de 2015 e que utilizassem o simulador *CloudSim* na validação das propostas.

3.1 *VBalance: A Selection Policy of Virtual Machines for Load Balancing in Cloud Computing*

Os servidores em um *data center* possuem recursos de hardware finitos. Logo, muitas vezes, um único servidor não é capaz de suprir as demandas de todas aplicações por ele hospedadas durante todo o período de execução. Por conta disso, é necessário realizar uma redistribuição dessas tarefas. A migração consiste em realocar uma Máquina Virtual do seu atual servidor para outro, mais apto a atender suas demandas. O primeiro passo desse processo é escolher a Máquina Virtual a migrada. Tal ação acontece através das Políticas de Seleção. Para resolver isso, Matos *et al.* (2015) propõe a Política de Seleção VBalance. Tal política, durante o processo de migração, busca selecionar a Máquina Virtual que mais contribui para o consumo de energia do servidor onde ela está hospedada. A proposta adota a Regressão Linear Múltipla (RLM) como ferramenta de apoio durante a tomada de decisão.

A Política VBalance, de início, elabora um histórico de informações sobre os servidores físicos as e máquinas virtuais. Periodicamente, são coletadas a energia consumida por cada um dos, no máximo, k servidores e o consumo de CPU de cada máquina virtual que ele hospeda. Quando um servidor é classificado como sobrecarregado, ou seja, está com uma elevada carga de trabalho, a Política de Seleção VBalance é chamada para escolher a melhor máquina virtual para migração. O critério adotado é analisar o nível de participação do consumo de CPU de cada VM

¹ <<https://ieeexplore.ieee.org/Xplore/home.jsp>>

² <<https://sucupira.capes.gov.br/sucupira/>>

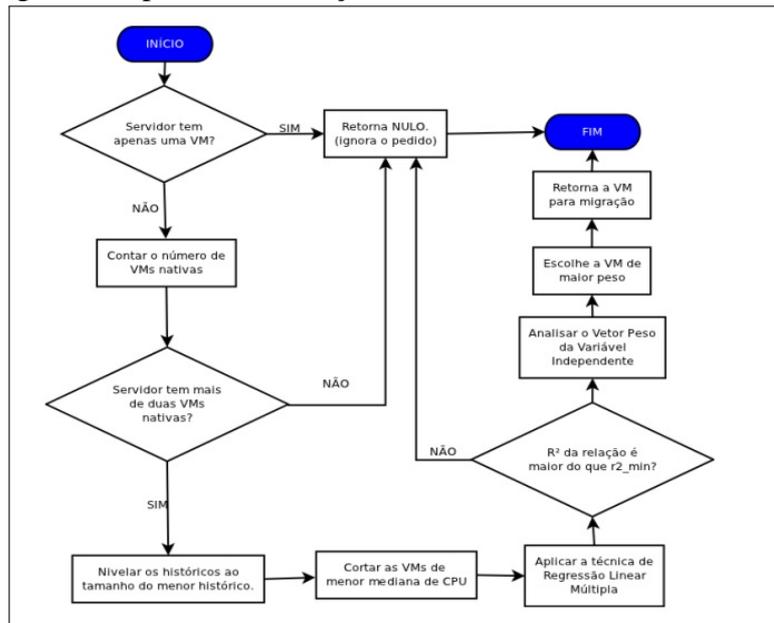
em relação ao consumo de energia de seu respectivo servidor. Para medir a relação, o modelo de Regressão Linear Múltipla é utilizado.

Após a coleta de informações e criação do histórico, a política está pronta para a tomada de decisão de escolher qual a Máquina Virtual mais apta a migração. Nessa fase, a coleção de dados levantada é enviada ao modelo de Regressão Linear Múltipla que retorna o nível de relação entre o consumo de CPU de cada VM com a energia consumida pelo seu servidor. Com base naquilo que é retornado pela RLM, a política pode escolher uma ou nenhuma máquina virtual para ser migrada. Esse último pode acontecer quando a política detecta que a qualidade da resposta da Regressão não é boa. Os históricos de CPU são reiniciados sempre que uma VM é migrada, pois o seu histórico consumo não possui relação com o consumo de energia do seu novo servidor. Isso pode produzir tamanhos diferentes de históricos, o que inviabiliza a RLM. Então, os autores dividiram as VMs em nativas e recém migradas. Uma Máquina Virtual é definida como nativa de um servidor quando houver pelo menos seis interações presentes no equipamento físico, caso contrário, a Máquina Virtual é marcada como recém migrada. Para o VBalance funcionar, o mesmo exige que haja pelo menos três máquinas nativas dentro do servidor sobrecarregado.

O modelo de Regressão Linear Múltipla exige que o número de variáveis e equações sejam as mesmas, o que pode não ocorrer devido aos diferentes tamanhos de históricos de CPU (devido as migrações). Para contornar isso, é calculado a mediana das coleções de dados de processador coletadas, e as que apresentarem históricos com valor menor que essa mediana, são excluídos da análise. Após essa etapa, o modelo de RLM receberá os dados que serão usados para calcular o Coeficiente de Determinação (CD). O Coeficiente de Determinação é uma métrica que é utilizada para medir a qualidade das previsões encontradas por um modelo de regressão em relação aos dados reais observados. Posteriormente, o resultado é comparado com um limiar mínimo (LM), que classifica o relacionamento como sendo bom ou ruim. Se o relacionamento é bom, então CD é maior que LM, então escolhe-se a Máquina Virtual de maior influência sobre o consumo de energia. Caso contrário, a política nega a escolha da Máquina Virtual baseada em uma relação fraca.

A política apresentada, baseado nos resultados nos testes, diminuiu consideravelmente o número de migrações de Máquinas Virtuais e proporcionou um bom nível de economia de energia ao *data center*. A contribuição de Matos et al. (2015) se deu na medida que optou por analisar a influência do consumo de CPU de Máquina Virtual na energia demandada pelo

Figura 9 – Fluxograma da política de seleção VBalance



Fonte: (Matos *et al.*, 2015)

servidor. A AHPSselection irá também analisar o consumo de CPU da Máquina Virtual como Matos *et al.* (2015) utilizou em seu trabalho.

3.2 *Energy-efficient and SLA-aware Virtual Machine Selection Algorithm for Dynamic Resource Allocation in Cloud Data Centers*

Segundo Moghaddam *et al.* (2018), o consumo de energia constitui uma proporção significativa dos custos operacionais dos *data centers*. Para isso, *data centers* realizam balanceamento de cargas para alocar eficientemente seus recursos sob demanda e otimizar o consumo de energia, selecionando Máquinas Virtuais a serem migradas. Esta migração pode culminar em uma baixa Qualidade de Serviço (QoS) e, conseqüentemente, uma maior Violação de Contratos de Nível de Serviço (SLAVs). Os autores apresentam um algoritmo que tem como principal critério analisar o nível de uso de CPU das Máquinas Virtuais em cada *host* e qualquer correlação entre elas.

Os autores determinam que migrar Máquinas Virtuais com alta correlação positiva de utilização de recursos de um servidor já sobrecarregado diminui a probabilidade deste mesmo servidor permanecer sobrecarregado na próxima iteração. Uma alta correlação positiva significa que duas ou mais Máquinas Virtuais, historicamente, possuem um mesmo comportamento de demanda por um determinado recurso, ou seja, a tendência é que todas elas aumentem e reduzam seu consumo por aquele recurso simultaneamente. Assim, para evitar que o servidor permaneça

sobrecarregado, é interessante que tais Máquinas Virtuais estejam em servidores diferentes e, para isso, algumas delas devem ser migradas.

No entanto, algumas vezes, Máquinas Virtuais são taxadas com alta utilização correlacionada de recursos, mas apresentam um uso pequeno de recursos do servidor. Nesse caso, migrar, não diminuiria significativamente o uso de recursos do servidor e o mesmo continuaria sobrecarregado. A técnica que Moghaddam *et al.* (2018) utilizou foi um método híbrido que considera a correlação de uso/utilização de recursos pelas Máquinas Virtuais e a proporção de recursos do servidor. O método produz uma lista de Máquinas Virtuais de acordo com o parâmetro de entrada n para o algoritmo e seleciona uma Máquina Virtual com o uso máximo de CPU dessa lista para migrar.

O algoritmo recebe como entrada um parâmetro n . Esse parâmetro definirá o tamanho da lista de máquinas virtuais a ser criada no algoritmo, nas quais serão selecionados as n máquinas virtuais com os n primeiros maiores coeficientes. O método proposto inicia construindo uma matriz utilizando as Máquinas Virtuais migráveis. A matriz é composta por l linhas e c colunas onde as linhas são os identificadores das Máquinas Virtuais aptas a migração e as colunas são os instantes de tempos onde foram coletadas informações sobre o consumo de CPU da respectiva máquina virtual. Cada elemento da matriz possui o valor de consumo do processador da Máquina Virtual v_l em um determinado instante de tempo t_c . O esquema da matriz sugerida é representada em 3.1.

$$A_{l,c} = \begin{pmatrix} v_1 t_1 & v_1 t_2 & \cdots & v_1 t_c \\ v_2 t_1 & v_2 t_2 & \cdots & v_2 t_c \\ \vdots & \vdots & \ddots & \vdots \\ v_l t_1 & v_l t_2 & \cdots & v_l t_c \end{pmatrix}. \quad (3.1)$$

Usando a matriz, o algoritmo computa os coeficientes de correlação de Pearson (SEDGWICK, 2012) entre todas as Máquinas Virtuais presentes na matriz. O coeficiente de correlação de Pearson é usado para calcular a correlação entre cada par de Máquinas Virtuais (que não estão em processo de migração) em cada intervalo de tempo. Para obter tal coeficiente, coleta-se as n últimas utilizações de CPU das duas Máquinas Virtuais em análise. Com base nos dados coletados, é calculada a média de utilização de CPU e, por fim, os valores obtidos são submetidos a uma fórmula proposta pelos autores.

Feito isso, a fórmula proposta calculará o coeficiente de correlação máxima para cada Máquina Virtual analisada. Em seguida, os coeficientes obtidos, e suas respectivas Máquinas

Virtuais, serão salvos em uma lista em ordem decrescente. Ao final, escolhe-se os n primeiros coeficientes de correlação máximo. Por fim, seleciona-se, desta lista, a Máquina Virtual com a maior utilização de CPU em um determinado tempo para migração.

O método garantiu bons resultados, principalmente na redução da Violação de Contratos de Nível de Serviço (SLAVs), quando as Máquinas Virtuais são altamente correlacionadas. O método proposto considerou somente a correlação linear entre Máquinas Virtuais. Para finalizar, Moghaddam *et al.* (2018) desafia investigar o efeito da correlação não linear para a Política de Seleção proposta e ainda os efeitos da correlação das Máquinas Virtuais ao selecionar *hosts* de destino para as mesmas. A AHPSelection apresentará abordagem parecida com a proposta do trabalho ao utilizar, como uma das métricas, a maior utilização de CPU para selecionar a máquina virtual.

3.3 *End-to-End informed VM Selection in Compute Clouds*

Segundo Teixeira e Bestavros (2015), a seleção de recursos, especialmente Máquinas Virtuais, geralmente é feita de maneira cega, uma vez que os usuários da Nuvem não possuem muitas informações sobre o consumo de recursos. Os padrões de comunicação, isto é, as diferentes formas de organizar os equipamentos de rede, desempenham papel importante na capacidade de resposta das aplicações em Nuvem sensíveis à latência. Melhor explicando, dependendo de como os equipamentos de rede são organizados em um *data center*, isto pode levar a um nível maior ou menor de atrasos e, por consequência, aumentar ou reduzir o nível de latência. Nesse trabalho, Teixeira e Bestavros (2015) discute um pouco sobre a latência de ponta a ponta do monitoramento de Máquinas Virtuais e discute também algoritmos de seleção de Máquinas Virtuais dentro da Nuvem, na qual, todos esses algoritmos usam algum tipo de informação de atraso para as sugestões de posicionamento das Máquinas Virtuais.

Nesse trabalho, Teixeira e Bestavros (2015) aborda algoritmos que utilizam informações de atraso em sua composição, são eles: Algoritmo de Colmeia (Hive Algorithm), Algoritmo Estrela (Star Algorithm), Algoritmo Melhores Centros (Best Centers Algorithm) e Algoritmos de Vizinhos (Neighbors Algorithm).

A Matriz de Latência é a base para os algoritmos de seleção de Máquinas Virtuais tratados nesse trabalho. Esta tem, como entrada primária, uma matriz contendo todas as Máquinas Virtuais e suas respectivas latências. Essa Matriz é constantemente atualizada com informações de ida e volta entre pares de Máquinas Virtuais. Cada linha ou coluna da matriz representa

uma Máquina Virtual na Nuvem e cada elemento representa o tempo de ida e volta entre duas Máquinas Virtuais. O autor optou por usar a latência como base para o algoritmo de seleção por que os RTTs (tempo de ida e volta) são métricas, simples, computacionalmente baratas e confiáveis para medir a distância da rede dentro das Nuvens.

O Algoritmo da Colmeia tem como objetivo encontrar um conjunto de Máquinas Virtuais, próximas umas as outras, com um tempo de ida e volta menor que um determinado parametro de entrada r (em milisegundos), ou seja, o RTT (tempo de ida e volta de um pacote) de qualquer par de Máquina Virtual pertencente a esse grupo deve ser menor que r . Esse algoritmo é muito difícil de implementar em termos práticos, pois é NP-difícil, ou seja, um caso complexo de resolução (Teixeira; Bestavros, 2015). As Máquinas Virtuais a serem migradas seriam aquelas que satisfazem a condição, na qual, o RTT é menor que o parâmetro r .

O Algoritmo Estrela é uma abordagem que visa organizar um conjunto de Máquinas Virtuais de acordo com uma Topologia em Estrela. Ao centro, é selecionada a Máquina Virtual que está mais próxima, em média, de todos os vizinhos. Para isso, busca-se a Máquina Virtual com a menor média de RTTs em relação a todas as outras Máquinas Virtuais no conjunto, ou seja, a que será escolhida como o centro da estrela. A Máquina Virtual escolhida para migração será aquela que o algoritmo indicar como o centro da topologia.

O Algoritmo Melhores Centros é uma generalização da abordagem do Algoritmo Estrela. Suponha que não se quer escolher apenas uma Máquina Virtual como centro da topologia em estrela, mas encontrar as N melhores Máquinas Virtuais como centros. O Algoritmo Melhores Centros realiza justamente essa escolha, também baseado nos RTTs. Neste caso, serão migradas todas as Máquinas Virtuais que estão no centro de alguma das estrelas encontradas.

O Algoritmo dos Vizinhos é um caso especial do Algoritmo de Colmeia, mas limitando o número de Máquinas Virtuais a apenas duas. Esse algoritmo tem como objetivo encontrar m pares de Máquinas Virtuais dentro da nuvem que estão muito próxima uma da outra, normalmente com um tempo de ida e volta entre eles abaixo de um dado limiar. Tal algoritmo recebe a Matriz de Latência como entrada e retorna todos os pares de Máquinas Virtuais cujo RTT é menor que o limite fornecido. Os pares escolhidos seriam direcionados para migração.

Segundo Teixeira e Bestavros (2015) estes métodos funcionam independentemente da arquitetura da nuvem, além de se adaptar a diferentes estações de trabalho, sendo leve e transparente. Segundo o mesmo autor, pesquisas recentes apontaram que padrões de comunicação são bastante relevantes, por isso, nesse trabalho, se concentraram nas políticas de seleção que

utilizam a latência das Máquinas Virtuais e contribuindo para a minimização do problema. A AHPSselection utiliza abordagem baseada em matrizes, assim como este trabalho apresenta, mas em vez de guardar as latências, a AHPSselection armazena os valores que quantificam a importância de um critério para outro ou alternativa para outro e vice-versa. Além disso, matrizes são computacionalmente mais rápidas, por isso, utilizar essa abordagem é interessante mas, tendo foco em melhorar a questão da energia.

3.4 Dynamic VM consolidation for energy-aware and SLA violation reduction in cloud computing

Neste trabalho, Cao e Dong (2012) afirma que, com a implantação em larga escala de *data centers*, o consumo de energia e a SLA já se tornaram uma questão urgente a ser resolvida. Segundo ele, é preciso projetar políticas tanto para reduzir o consumo de energia, quanto para diminuir as violações de SLA. Para isso, Cao e Dong (2012) desenvolveu uma Política de Seleção de Máquinas Virtuais que utiliza a média e o desvio padrão para definir se o servidor está sobrecarregado ou não e o coeficiente de correlação máxima positivo é usado para selecionar as Máquinas Virtuais que sobrecarregam o servidor para migração.

Primeiramente, Cao e Dong (2012) utiliza-se de uma estratégia para detectar os servidores sobrecarregados. De início, é definido um parâmetro seguro. Tal parâmetro é calculado para cada servidor do *data center* que se deseja saber se está ou não sobrecarregado. O cálculo tem como base a análise do histórico de consumo de CPU do servidor e das Máquinas Virtuais por ele hospedadas.

O primeiro passo é, dado um intervalo de tempo, coletar os valores de utilização máxima de CPU de cada Máquina Virtual e do servidor físico em questão. Os autores propuseram uma fórmula para detectar se o *host* está sobrecarregado ou não. Se o resultado da fórmula for maior que a situação de utilização atual do processador, então a máquina está sobrecarregada. Se não, a máquina não está sobrecarregada.

Para a Política de Seleção de Máquinas Virtuais, os autores utilizaram uma política já existente, mas melhorando-a e a tornando mais eficiente. A política apresentada por ele é uma extensão da Política de Máxima Correlação baseado na ideia de (VERMA *et al.*, 2009). Tal política se baseia em escolher a Máquina Virtual que tem o maior coeficiente de correlação máxima em relação as demais hospedadas naquele servidor. Esse processo é repetido até que o servidor não esteja mais sobrecarregado.

O coeficiente de correlação é calculado por uma fórmula que utiliza, como parâmetro, o consumo de CPU da Máquina Virtual em análise e a soma de todas as outros presentes no servidor. Considere, como exemplo, três Máquinas Virtuais: $V1, V2$ e $V3$. Como entrada, é dado uma lista de tamanho n contendo as utilizações de processador das Máquinas Virtuais. No exemplo, $n = 3$. Primeiro, toma-se uma variável como dependente para ser analisada, por exemplo, a variável de utilização relacionada a $V1$. Após isso, toma-se todas as outras variáveis como independentes, por exemplo, as variáveis de utilização relacionada a $V2$ e $V3$. O objetivo é avaliar a correlação entre a variável dependente e as variáveis independentes, ou seja, medir as relações entre essas variáveis. Para melhor compreender, a correlação procura entender como uma variável se comporta em um cenário quando outra variável está variando. Isso ajuda identificar o grau de correlação de cada variável, que nesse contexto representa cada Máquina Virtual. Esse mecanismo é feito tomando cada VM do grupo como variável dependente e as demais como independentes. Após esses procedimentos, é escolhido a Máquina Virtual dependente que apresentou o maior valor de coeficiente de correlação.

O autor aprimorou esse método utilizando a estatística para melhorar até então o método citado anteriormente e construiu um tipo de heurística. A melhoria parte da informação de que o intervalo do coeficiente de correlação está entre -1 e 1 . Quando o intervalo está entre -1 e 0 , é chamado de correlação negativa. Caso contrário, ou seja, entre 0 e 1 , é chamado de correlação positiva. Segundo os próprios autores, a correlação negativa indica que duas variáveis raramente aumentam ou diminuem no mesmo momento. Em contraste, a correlação positiva indica que as variáveis aumentam ou diminuem simultaneamente na maioria dos casos. Portanto, se a correlação for positiva, o servidor é mais propenso a ter sobrecarga do que o negativo.

O autor afirma que, baseados nas experiências feitas, as políticas, principalmente a política de seleção, tiveram um desempenho muito superior comparado a trabalhos anteriores citados por ele. Mas, no quesito consumo de energia, o mesmo, não conseguiu ter uma diminuição esperada. Um dos pontos fracos desse trabalho é justamente o consumo energético. A AHPSselection, por sua vez, é estruturada para reduzir, de maneira significativa, o consumo de energia.

3.5 Fechamento do Capítulo

Para finalizar este Capítulo, segue um breve resumo com os principais pontos comparativos entre os Trabalhos Relacionados aqui apresentados e a Política de Seleção de Máquinas

Virtuais proposta neste trabalho (AHPSelection):

1. Matos *et al.* (2015) utilizou somente o processador como critério, já a AHPSelection utiliza multicritérios para a escolha. Além disso, a VBalance não foi testada em conjunto com Políticas de Alocação desenvolvidas com o propósito de economizar energia. Um das Políticas de Alocação que a AHPSelection irá atuar será a MSI que tem o foco na energia.
2. O trabalho de Moghaddam *et al.* (2018) não apresentou resultados significativos em todos os *traces* no qual foi testada, o que representa uma potencial limitação nesse quesito. O objetivo principal da AHPSelection é a economia de energia. Além disso, a AHPSelection é multicritério sendo que no trabalho de Moghaddam *et al.* (2018) foi utilizado um critério somente.
3. O trabalho de Teixeira e Bestavros (2015) não teve um foco na energia. A AHPSelection, por sua vez, prioriza a economia de energia, pois é um dos problemas mais pertinentes em um ambiente de *data center* como já explanado anteriormente e, também, é multicritério.
4. Cao e Dong (2012) não conseguiu reduzir significamente e obter resultados satisfatórios quanto a energia. Já a AHPSelection tem foco na energia baseado em diversos critérios.

4 AHPSELECTION

Neste capítulo será apresentada a Política de Seleção AHPSelection que utiliza o método AHP no procedimento de escolha de uma ou mais Máquinas Virtuais para serem migradas de um servidor sobrecarregado para outro com menor carga de trabalho. Ressalta-se, mais uma vez, que a AHPSelection executa apenas a escolha da Máquina Virtual, cabendo a Política de Alocação escolher o servidor destino e gerenciar o processo de migração.

4.1 AHP e a Seleção de Máquinas Virtuais

A escolha de uma ou mais Máquinas Virtuais para migração pode ser vista como uma tomada de decisão. Um dos métodos de tomada de decisão escolhidos para este trabalho foi o método AHP. Um dos motivos de utilizar a técnica AHP é a sua simplicidade e sua popularidade no meio acadêmico (HO, 2008). Segundo, Carli *et al.* (2010), a AHP vem sendo bastante utilizada em problemas relacionados a tomada de decisão multicritério. Além disso, um dos principais modelos matemáticos que são usados para realizar escolhas, principalmente no mercado, utiliza a AHP (VARGAS; PMP, 2010). E por se tratar de um método que avalia diversos critérios, a AHP foi escolhida como método para auxiliar na escolha da Máquina Virtual a ser migrada.

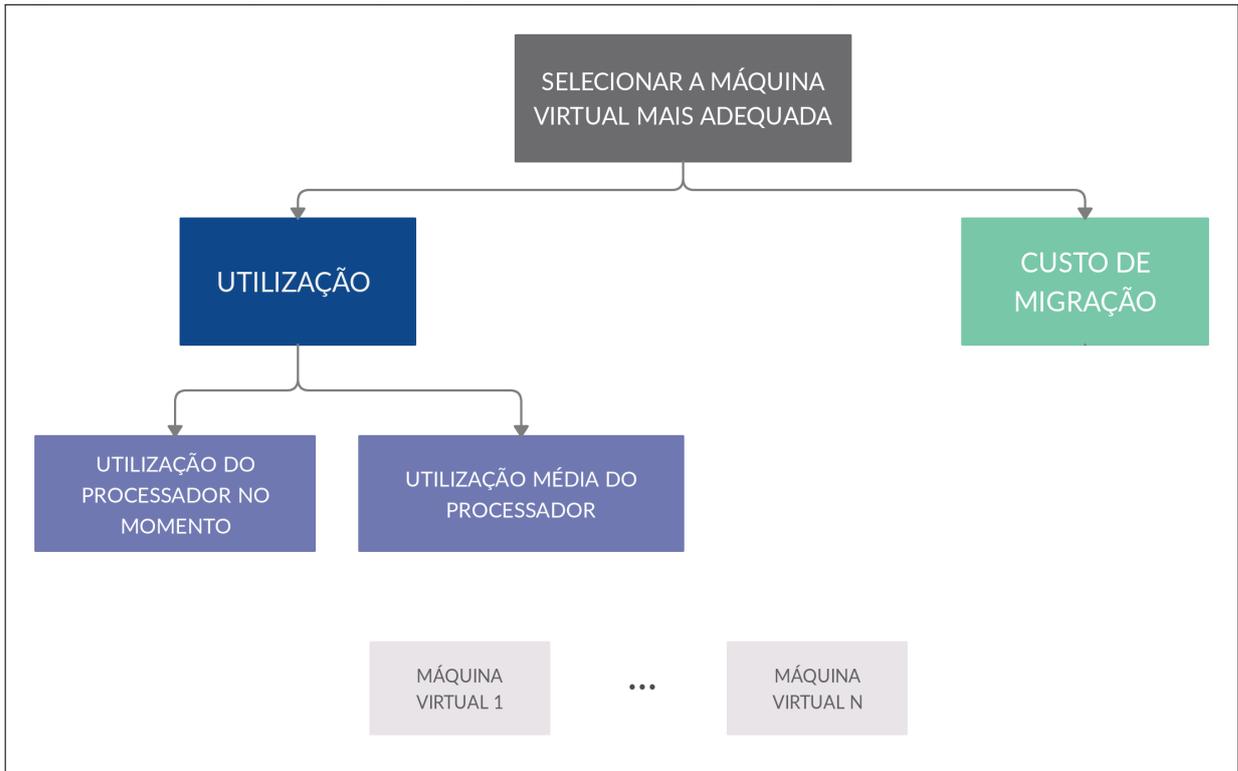
Primeiramente, como visto anteriormente, para se construir uma AHP é preciso definir o problema e o tipo de conhecimento que se quer obter. O problema é definido como a escolha da(s) melhor(es) Máquinas Virtuais de um servidor sobrecarregado. Um servidor está sobrecarregado quando o próprio está no limite de uso de seus recursos computacionais, ou seja, está com o consumo de um ou mais recursos (CPU, Memória Primária, Disco e/ou Largura de Banda) acima de um determinado limiar superior, ou seja, quando atinge 100% da sua capacidade total.

Feito isso, o próximo passo é determinar o tipo de conhecimento que se objetiva atingir. Nesse caso, o tipo de conhecimento é encontrar a(s) Máquina(s) Virtual(is) mais adequada(s) para migração dado alguns critérios. A Máquina Virtual será considerada adequada para migração atendendo alguns critérios. O primeiro critério é o relacionado a Utilização do Processador. A Máquina Virtual a ser migrada apresenta menor consumo de momento do processador e possui uma média de utilização do processador menor. Outro critério é o de Custo de Migração, onde será selecionada a Máquina Virtual que consome menos memória. Estes parâmetros servirão de base para definir a lista de critérios e subcritérios a serem analisados no

processo de tomada de decisão.

Definidos o problema e o tipo de conhecimento, a próxima etapa é construir a hierarquia AHP baseado no modelo de Saaty (2008). A hierarquia conterá o objetivo (já definido no tipo de conhecimento), os critérios, os subcritérios e as opções de escolha ou alternativas. A hierarquia final proposta para este problema é apresentado na Figura 10.

Figura 10 – Hierarquia AHP para o problema de seleção de máquinas virtuais



Fonte: Próprio autor

Para este problema, dois critérios foram tomados como base para auxiliar na tomada de decisão: a Utilização do Processador e o Custo de Migração. Esses critérios foram escolhidos pois impactam no consumo energético. A escolha desses critérios foi o fato de que o processador e a memória são um dos componentes que mais consomem energia em uma máquina.

Cada critério possui um conjunto de subcritérios associados. Para analisar a Utilização do Processador foi considerado captar informações dos componentes da Máquina Física. Portanto, a Utilização do Processador tem como subcritérios: a Utilização do Processador no Momento da tomada de decisão com base no histórico de consumo e a Utilização Média do Processador pela Máquina Virtual. Estes são levados em consideração pois são os componentes principais da máquina física e que mais consomem energia.

Por último, o critério de Custo de Migração. Um dos componentes responsáveis pelo

custo migratório é o Random Access Memory (RAM). Dessa maneira, foi sugerido o Consumo de RAM como um subcritério do Custo de Migração.

Para encerrar a hierarquia, no nível mais baixo, é preciso definir as alternativas/opções. Para o problema serão definidos como alternativas/opções de escolha as n Máquinas Virtuais presentes no servidor sobrecarregado.

Após isso, é o momento de julgar e ponderar critérios, subcritérios a alternativas organizando-as em uma estrutura de matrizes. Para isso, constrói-se matrizes de comparação onde as linhas e as colunas referenciam os critérios, subcritérios ou alternativas. Na próxima seção será mostrado como foram criadas e valoradas matrizes usadas pela política proposta neste trabalho.

4.2 Matrizes de Comparação

A construção dessas matrizes de comparação dos critérios ocorreu de maneira empírica. Primeiramente, foi definido que o critério de Utilização do Processador seria de igual importância em relação ao Custo de Migração. Utilizando a tabela de (SAATY, 2008) foi definido, de forma empírica, que a Utilização do Processador teria importância 1 em relação ao Custo de Migração. Logo, o Custo de Migração foi definido como tendo importância 1 em relação a Utilização do Processador. Os dois critérios foram ponderados iguais para que possa existir uma equivalência entre a energia e o desempenho, já que a diminuição de energia pode causar violações SLA. O esquema da matriz de comparação dos critérios para este problema é representado no Quadro 1.

Quadro 1 – Tabela que esquematiza a matriz de comparação dos critérios

	Utilização do Processador	Custo de Migração
Utilização do Processador	1	1
Custo de Migração	1	1

Fonte: elaborado pelo autor.

O mesmo é feito para os subcritérios de cada critério definido e para as alternativas. Para o subcritério da RAM, por se tratar somente de um subcritério, sua matriz é 1.

Para os subcritérios do critério de Utilização do Processador foi definido, baseada em testes, que a Utilização do Processador no Momento teria importância $\frac{1}{2}$ em relação a Utilização Média do Processador. Logo, a Utilização Média do Processador foi definida com prioridade 2 em relação a Utilização do Processador no Momento. A representação é dada pelo Quadro 2.

Quadro 2 – Tabela que esquematiza a matriz de comparação dos subcritérios do critério de Utilização do Processador

	Utilização do Processador no Momento	Utilização Média
Utilização do Processador no Momento	1	$\frac{1}{2}$
Utilização Média	2	1

Fonte: elaborado pelo autor.

Feito isso, é preciso ponderar as alternativas em relação aos critérios e subcritérios. Para isso, foi criado um algoritmo para relacionar as diferentes alternativas para cada critério. A próxima subseção aborda como funciona esse algoritmo.

4.2.1 Algoritmo para a ponderação das alternativas

Para construir a matriz de comparação das alternativas é preciso ponderá-las baseado na tabela de Saaty (2008). Como os valores das alternativas serão coletados pelo algoritmo em tempo de execução, não é possível definir as ponderações das alternativas previamente. Logo, é preciso utilizar um algoritmo para realizar essa tarefa. O algoritmo recebe como entrada os valores X , Y e um máximo MAX , e como saída um valor que está no intervalo da tabela de Saaty (2008). A ideia geral do Algoritmo é relacionar a importância do valor X frente ao valor Y dado que o valor máximo que ambos podem assumir é MAX . Por exemplo, dado uma Utilização do Processador no Momento de valor X de uma Máquina Virtual A e uma Utilização do Processador no Momento de valor Y de uma Máquina Virtual B , esse algoritmo retorna um valor da tabela proposto por Saaty (2008) relacionando a importância de X frente a Y que vai de 1 a 9 ou $\frac{1}{2}$ a $\frac{1}{9}$. O pseudocódigo desse algoritmo é apresentado no Algoritmo 1.

Algoritmo 1: Algoritmo para a ponderação das alternativas

```

1: Entrada:  $X, Y, MAX$ 
2: Saída: Valor entre 1 e 9 ou  $\frac{1}{2}$  e  $\frac{1}{9}$ 

3:  $RES \leftarrow X - Y + MAX$ 

4: se  $RES > MAX$  então
5:   para  $i = 1$  até 8
6:     se  $RES < (MAX + (i * MAX / 8))$  então
7:       Retorne  $i + 1$ 

8: se  $RES < MAX$  então
9:   para  $i = 1$  até 8
10:    se  $res < (i * MAX / 8)$ 
11:      Retorne  $1 / (10 - i)$ 

12: Retorne 1

```

O Algoritmo 1 inicia recebendo como entrada três valores: o valor X , Y e MAX . Dado isso, na linha 3, calcula-se a diferença de X e Y somado com o valor MAX . Na linha 4 temos que se o valor $RES > MAX$ então RES tem uma prioridade alta. As linhas 5 a 7 calculam essa prioridade. Se não, na linha 8, o $RES < MAX$, então sua prioridade é mais baixa. Da linha 9 a 11 é calculado essa prioridade. Agora, se $RES = MAX$, então retorna como prioridade o valor 1, que significa na tabela de Saaty (2008), a igualdade. Para melhor compreensão, será utilizado um exemplo de resposta desse algoritmo.

Suponha uma Máquina Física que possui a Máquina Virtual A e a Máquina Virtual B. Assuma que a Máquina Virtual A possui uma Utilização do Processador de 0,6 e a Máquina Virtual B possui uma Utilização de 0,4. O valor máximo de Utilização do Processador é 1, ou seja, 100% da capacidade. Dessa maneira, o Algoritmo 1, receberia como entrada o $X = 0,6$, $Y = 0,4$ e o $MAX = 1$. Primeiro, é preciso calcular o RES .

$$RES \leftarrow X - Y + MAX \quad (4.1)$$

$$RES \leftarrow 0,6 - 0,4 + 1 \quad (4.2)$$

$$RES \leftarrow 1,2 \quad (4.3)$$

Note que $RES > MAX$, portanto entra na linha 4 do Algoritmo 1. Após as interações é retornado que a ponderação de X em relação a Y é 3, pois na linha 6, o RES fica menor quando o $i = 2$. Logo, a ponderação da Máquina Virtual A em relação a Máquina Virtual B é 3.

É possível notar que esse algoritmo realiza estimativas quanto a ponderação de X em relação a Y . Esse fato pode ocasionar inconsistências, às vezes. A AHPSelection utiliza a mesma estratégia de verificação de consistência elaborado por Saaty (1984) e já explicado na subseção 2.6.1. Se o valor da Razão de Consistência (RC) for maior ou igual a 0.1, então a AHPSelection decide não prosseguir e nega escolher uma Máquina Virtual, visto que a matriz está inconsistente. Caso contrário, a AHPSelection continua o processo a fim de escolher uma Máquina Virtual, já que a matriz está consistente.

Feito isso, com base no método proposto por Saaty (2008) e já explicado na Seção 2.7, após esse passo, são calculadas as prioridades. Agora, basta realizar o restante do procedimento, relacionando as alternativas com os critérios e os subcritérios, obter a Prioridade Global de cada uma delas e a que estiver com maior prioridade, será a escolhida como alvo.

Feito isso, é preciso analisar a validade e eficiência da política proposta nesse trabalho, a AHPSelection. Nas próximas seções, serão mostrados o ambiente e cenário de testes, os resultados obtidos e as conclusões.

5 AMBIENTE DE TESTES

Após a apresentação da proposta da política de seleção de máquinas virtuais AHPSelection, com seus critérios e subcritérios definidos, é preciso validar sua eficiência em relação as demais políticas já existentes. Para isso foi utilizado o simulador *CloudSim* (CALHEIROS *et al.*, 2011) em sua versão atualizada *CloudSim 3.0.3*. Essa ferramenta é ideal para simular ambientes de Computação em Nuvem contendo a arquitetura de um ambiente real convencional como servidores, máquinas virtuais, políticas de seleção, políticas de alocação, dentre outros.

Visando uma maior confiabilidade nos resultados obtidos, é interessante que os cenário simulado sejam semelhantes a ambientes reais de Nuvem. Por este motivo, foi escolhido parte do cenário utilizado por Mann e Szabó (2017) para validar o desempenho da política proposta nesse trabalho. Mann e Szabó (2017) utilizou as mesmas configurações e métricas adotadas por Beloglazov e Buyya (2012b) e adicionou algumas novas configurações e *traces* de dados ao cenário original.

Nas próximas sessões serão detalhados o cenário e as configurações gerais utilizadas, as métricas que serão utilizadas para a validação e análise do comportamento das políticas e, por fim, os resultados encontrados.

5.1 Configuração Geral do Cenário

Para que o resultado fosse próximo da realidade, foram definidos especificações próximas de um *data center* convencional. O *data center* foi configurado com 800 servidores, onde metade desses são do tipo HP ProLiant G4 e a outra metade são do tipo HP ProLiant G5. De uma forma geral, as máquinas possuem as mesmas especificações de *hardware*, o que diferencia é a capacidade máxima de processamento por núcleo. A Tabela 6 mostra as especificações dos servidores utilizados.

Tabela 6 – Especificação dos servidores utilizados

Recurso	HP ProLiant G4	HP ProLiant G5
Número de Núcleos	2	2
Processamento por núcleo	1860 MIPS	2660 MIPS
Memória Primária	4 GB	4 GB
Memória Secundária	1 TB	1 TB
Largura de Banda	1 Gbit/s	1 Gbit/s
Qtde de servidores	400	400

Fonte: (BELOGLAZOV; BUYYA, 2012a)

Além de definir as configurações de *hardware*, também é importante definir o padrão de consumo de energia de cada tipo de servidor. O cálculo do consumo energético é obtido a partir da carga de trabalho de CPU em um determinado período. Para este trabalho, foi considerado que o servidor que atinja 80% ou mais de sua capacidade total será considerado sobrecarregado. A Tabela 7 mostra a relação da carga com o consumo. Vale ressaltar que os valores intermediários aos percentuais apresentados na Tabela 7 são obtidos através da interpolação entre os percentuais (mínimo e máximo) mais próximos a ele.

Tabela 7 – Consumo de energia por carga de trabalho

Servidor	0%	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
HP G4 (W)	86	89.4	92.6	96	99.5	102	106	108	112	114	117
HP G5 (W)	93.7	97	101	105	110	116	121	125	129	133	135

Fonte: (BELOGLAZOV; BUYYA, 2012a)

As Máquinas Virtuais também herdam configurações de um cenário real. Foram utilizadas quatro instâncias de Máquinas Virtuais disponibilizadas pela empresa Amazon através do serviço Amazon EC2¹: Microinstância (*Micro Instance*), Instância Pequena (*Small Instance*), Instância Extra Grande (*Extra Large Instance*) e Instância Média Otimizada Para Computação (*High-CPU Medium Instance*). A Tabela 8 mostra as especificações dessas máquinas virtuais. Note que, a principal diferença de configuração entre os tipos de máquinas virtuais são as quantidades máximas alocadas de processamento e de memória primária, o que contribui para a proposta da AHPSelection de ponderar apenas estes recursos como critérios para a tomada de decisão.

Tabela 8 – Especificação das Máquinas Virtuais

Recurso	High-CPU Medium	Extra Large	Small	Micro
Qtde. de núcleos	1	1	1	1
Processamento	2500 MIPS	2000 MIPS	1000 MIPS	500 MIPS
Memória Primária	870 MB	1.7 GB	1.7 GB	613 MB
Tamanho da máquina virtual	2.5 GB	2.5 GB	2.5 GB	2.5 GB
Largura de banda	10 Mbits/s	10 Mbits/s	10 Mbits/s	10 Mbits/s

Fonte: Adaptada de (BELOGLAZOV; BUYYA, 2012a)

Uma entidade importante do simulador é a *Cloudlet*, cada instância de *Cloudlet* tem a função de gerenciar a execução de uma tarefa, como determinar a quantidade de recursos requisitados por ela em um determinado momento (CALHEIROS *et al.*, 2011). Cada tarefa é representada por um *trace* de dados obtido de um cenário real de Nuvem. No cenário cada

¹ <<http://aws.amazon.com/pt/ec2/>>

Cloudlet é executada por uma única máquina virtual. No cenário são utilizadas cargas de *traces* distintas. Abaixo, serão descritos com um pouco mais de detalhes estes *traces*.

5.1.1 Cenário (proposto por (MANN; SZABÓ, 2017))

As tarefas (carga de trabalho) deste cenário possuem três fontes oriundas de ambientes reais: PlanetLab, Bitbrains e Google Cluster.

Os dados obtidos através do PlanetLab retratam a carga de trabalho de 1052 máquinas virtuais durante 1 dia. Já os obtidos através do Google Cluster retratam a carga de trabalho de 12000 servidores durante 29 dias. Os dados obtidos através do provedor BitBrains são de 1750 máquinas virtuais em um período de 4 meses. Nas três fontes citadas o período de amostragem se deu a cada 5 minutos.

A Tabela 9 apresenta algumas das especificações dos ambientes onde os *traces* foram obtidos. Cada *trace* possui sua peculiaridade quanto ao rastreamento das cargas de trabalho. Todos os *traces* possuem um rastreio em comum que é o processador. Nos outros casos, apenas alguns fazem o rastreio da memória e disco. Todas as características desses três ambientes, em relação ao rastreamento ou não da CPU, memória ou disco são descritas na Tabela 9.

Tabela 9 – Especificação das Máquinas Virtuais

Origem	Virtualizado	CPU	Memória	Disco
PlanetLab	sim	sim	não	não
Google	não	sim	sim	sim
Bitbrains	sim	sim	sim	não

Fonte: Adaptada de (MANN; SZABÓ, 2017)

5.2 Métricas

Dado o cenário de teste, é preciso definir os dados a serem coletados para análise e validação da política. Este trabalho irá utilizar as mesmas métricas usadas por Mann e Szabó (2017) acrescentada da métrica referente a quantidade de migrações realizadas durante a simulação.

Além da métrica do consumo de energia por dia simulado, também estão presentes métricas referentes ao desempenho do *data center*: a *Performance Degradation due to Migration* (PDM), que mede a violação de SLA causada pelas migrações, a *SLA Violation Time per Active Host* (SLATAH), que mede a violação causada pela utilização total do servidor e, por fim, a

quantidade de migrações efetuadas durante a simulação. Nas sessões seguintes serão detalhadas cada uma dessas métricas.

5.2.1 *Consumo de energia*

Durante o tempo de operação do *data center*, é preciso energia elétrica para alimentar os servidores que abrigam as Máquinas Virtuais. Essa métrica mensura o total de energia consumido pelo *data center* por período simulado. O cálculo desta métrica está diretamente relacionado a Tabela 7, pois é a partir dela que se obtém o consumo momentâneo de energia de cada servidor e, por consequência, o consumo de todos os servidores durante a simulação. Está é a principal métrica, pois o objetivo central deste trabalho é propor uma política que diminua o consumo de energia do *data center*.

5.2.2 *Quantidade de migrações*

Durante a execução de tarefas pelas máquinas virtuais, devido a elasticidade oferecida pela Virtualização, as máquinas virtuais podem exigir mais ou menos recursos do servidor (como quantidade de CPU ou de Memória RAM a elas alocadas por exemplo). Tais mudança podem ocasionar em sobrecarga ou em subutilização do servidor, situações indesejadas sob a ótica de violações de SLA ou de consumo de energia, respectivamente. A referida métrica apresenta a quantidade de migrações realizadas por dia simulado.

5.2.3 *Performace Degradation due to Migration*

Para migrar uma Máquina Virtual é preciso o uso de parte de seus recursos computacionais alocados para executar a migração. Isso pode gerar uma violação de SLA, pois pode ocorrer de uma Máquina Virtual não ter todos os recursos solicitados por ela durante aquele período específico. O *CloudSim* representa esta violação através da métrica *Performace Degradation due to Migration* (PDM). Tal métrica calcula, em porcentagem média, a quantidade de MIPS indisponível às máquinas virtuais durante a migração. A Fórmula 5.1 mostra matematicamente como a PDM é calculada.

$$SLAPDM = \frac{1}{M} \sum_{i=1}^M \frac{C_{dj}}{C_{rj}} \quad (5.1)$$

Onde:

M = Número de máquinas virtuais

C_{dj} = CPU não utilizada devido a migração

C_{rj} = CPU solicitada durante a migração

5.2.4 SLA Violation Time per Active Host

Quando um servidor utiliza 100% de sua capacidade de processamento, é nítido que nenhuma de suas máquinas virtuais hospedes poderá solicitar mais recursos de CPU. Se alguma delas solicitar um MIP a mais, isso ocasionará uma violação de SLA, já que os recursos estão esgotados e não é possível atender a requisição da Máquina Virtual. O *CloudSim* representa essa violação através da métrica *SLA Violation Time per Active Host* ou SLATAH. Esta métrica calcula o percentual médio de tempo que os servidores, enquanto ativos, estiveram com 100% de consumo de processamento. A Fórmula 5.2 demonstra matematicamente como essa métrica é calculada.

$$SLATAH = \frac{1}{N} \sum_{i=1}^N \frac{T_{si}}{T_{ai}} \quad (5.2)$$

Onde:

N = Número de máquinas virtuais

T_{si} = Tempo que o servidor i está utilizando toda CPU

T_{ai} = Tempo que o servidor i está ativo

5.3 Políticas de Seleção e Alocação

Para validar a política de seleção AHPSelection proposta nesse trabalho, é preciso confrontar sua eficiência considerando outras políticas já existentes na literatura. Para isso, foram utilizadas quatro outras políticas de seleção, todas já disponibilizadas pelo simulador *CloudSim* e oriundas de Beloglazov e Buyya (2010) e Beloglazov e Buyya (2012c), são elas:

- **Random Selection (RS):** Escolhe aleatoriamente uma Máquina Virtual dentre todas aquelas hospedadas pelo servidor sobrecarregado.
- **Maximum Correlation (MC):** Essa política se baseia na premissa de que quanto maior a correlação entre o uso dos recursos, maior a probabilidade do servidor se encontrar sobrecarregado. A ideia é selecionar as Máquinas Virtuais para migração que possuem maior correlação da utilização da CPU em relação as outras.

- **Minimum Migration Time (MMT):** A ideia dessa política é escolher uma Máquina Virtual para migração que necessita do tempo mínimo para concluir uma migração em relação às outras. O tempo é calculado dividindo a quantidade de memória RAM pela quantidade de largura de banda de rede.
- **Minimum Utilization (MU):** Essa política baseia sua escolha em minimizar o aumento potencial total da utilização e violação do SLA. Para isso, essa política escolhe a Máquina Virtual que tem o menor uso de CPU em relação às solicitadas.

Por fim, como já descrito nesse trabalho, uma Política de Seleção precisa trabalhar em conjunto com uma Política de Alocação, pois é necessário que a máquina virtual escolhida pela primeira seja realocada em outro servidor e, a escolha deste servidor destino, é de responsabilidade da Política de Alocação. Assim, embora não seja o foco deste trabalho, se faz necessário a adoção de algumas Políticas de Alocação para que o cenário seja executado. As Políticas escolhidas se basearam no trabalho de Guazzone *et al.* (2012), Beloglazov e Buyya (2012b) e (FILHO, 2018). Tais políticas foram escolhidas por serem aquelas que obtiveram melhor (MSIAllocator e Guazzone) e pior desempenho (*Migration Static Threshold*) em relação ao consumo de energia no referido trabalho. Logo abaixo, essas políticas são melhor descritas.

- **Guazzone (GUA):** Esta política utilizou uma heurística BFD (*Best Fit Decreasing*) para definir a alocação das Máquinas Virtuais. Nessa política, os servidores ligados tem prioridade e são classificados quanto a capacidade de CPU livre em ordem decrescente.
- **Migration Static Threshold (THR):** Essa política sugere realocar máquinas virtuais de servidores que estejam sobrecarregados. Nessa proposta, é verificado o tempo de migração das Máquinas Virtuais e enquanto o servidor estiver sobrecarregado, as Máquinas Virtuais sofrerão migrações.
- **MSIAllocator (MSI):** Essa política utiliza a meta-heurística *Iterated Local Search* (ILS) para realizar a alocação das Máquinas Virtuais determinando, através dessa meta-heurística e do consumo de CPU, o posicionamento das Máquinas Virtuais.

5.4 Configuração do ambiente de teste

Para fins de registro, a simulação foi executada em uma máquina com processador Intel(R) Core(TM) i5-6200U CPU @ 2.30GHz com arquitetura 64-bit, memória RAM DDR3 de 8 GB com velocidade de clock 1600 MHz, HD de 1 TB de capacidade e placa de vídeo Intel HD graphics 520 integrada. O Sistema Operacional utilizado é o Ubuntu versão 16.04. Todos os

testes foram realizados nesta mesma máquina.

6 RESULTADOS

Nesta seção, serão mostrados os resultados dos testes do cenário. O resultados serão analisados, de início, por trace (*PlanetLab, BitBrains, GoogleCluster*). Em cada análise, por sua vez, serão avaliadas as métricas pré-estabelecidas (consumo de energia, quantidade de migrações, PDM e SLATAH) separadamente. Os testes foram executados utilizando as Políticas de Alocação e de Seleção presentes na Tabela 10. As Políticas de Seleção escolhidas se basearam no fato de que todas elas já se encontram implementadas no simulador *CloudSim*, além de serem bem aceitas dentro do campo acadêmico.

Tabela 10 – Políticas de Alocação e Seleção adotadas

Nome da política	Sigla	Referência
Guazzone	Guazzone	(GUAZZONE <i>et al.</i> , 2012)
Migration Static Threshold	THR	(BELOGLAZOV; BUYYA, 2012b)
MSIAllocator	MSI	(FILHO, 2018)
Random Selection	RS	(BELOGLAZOV; BUYYA, 2012c)
Maximum Correlation	MC	(BELOGLAZOV; BUYYA, 2012c)
Minimum Migration Time	MMT	(BELOGLAZOV; BUYYA, 2012c)
Minimum Utilization	MU	(BELOGLAZOV; BUYYA, 2010)

Fonte: Adaptado de (SAATY, 2008)

Será observado posteriormente que, os resultados dos testes mostraram que a Política AHPSelection apresentou um certo equilíbrio entre o consumo de energia e as métricas relacionadas ao desempenho. Tal comportamento é motivado pela estruturada hierárquica definida pela AHP, de um lado o critério de Utilização do Processador, que está ligado ao consumo de energia, do outro o Custo de Migração, relacionado a memória principal e intimamente ligado as métricas de desempenho avaliadas (número de migração, PDM e SLATAH). Isso fez com que a política apresentasse, de maneira tímida, uma economia de energia ao passo que manteve um bom desempenho. Este resultado já mostra um bom comportamento da AHPSelection, contudo, como o objetivo do trabalho é minimizar o consumo de energia, pensou-se em uma pequena modificação na proposta inicial visando atingir este objetivo.

De acordo com os resultados obtidos por Matos *et al.* (2015), é possível concluir que, em algumas situações, negar a escolha de uma máquina virtual para migração pode representar economia de energia, pois diminui a quantidade de migrações e consequentemente diminui o consumo de energia. Com base nesta conclusão, foi criada um nova versão da política: a AHPSelection2. Esta última, funciona da mesma maneira que a original: mesmos critérios, pesos e estrutura hierárquica. Entretanto, o momento da decisão de quando migrar é o que as

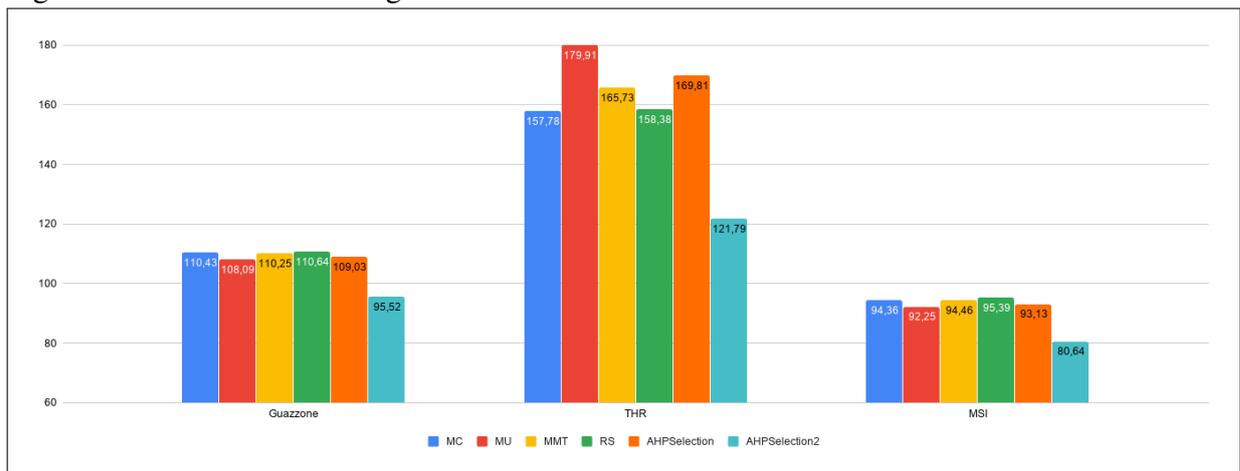
diferencia: enquanto a versão inicial da política escolhe a máquina virtual para migração no mesmo instante da solicitação, a AHPSelection2 só atua quando o servidor está sobrecarregado a, pelo menos, um período, ou seja, ela adia o processo de migração em, pelo menos, um turno. Nas sessões seguintes, serão mostrados e analisados os resultados obtidos por cada política.

6.1 Trace PlanetLab

Nesta seção será apresentado os resultados dos desempenho das políticas de seleção com cada Política de Alocação utilizando o *trace PlanetLab*. Em cada subseção será abordado os resultados de acordo com cada métrica já estabelecidas na Seção 5.2.

6.1.1 Consumo de energia

Figura 11 – Consumo de energia do trace PlanetLab



Fonte: Próprio autor

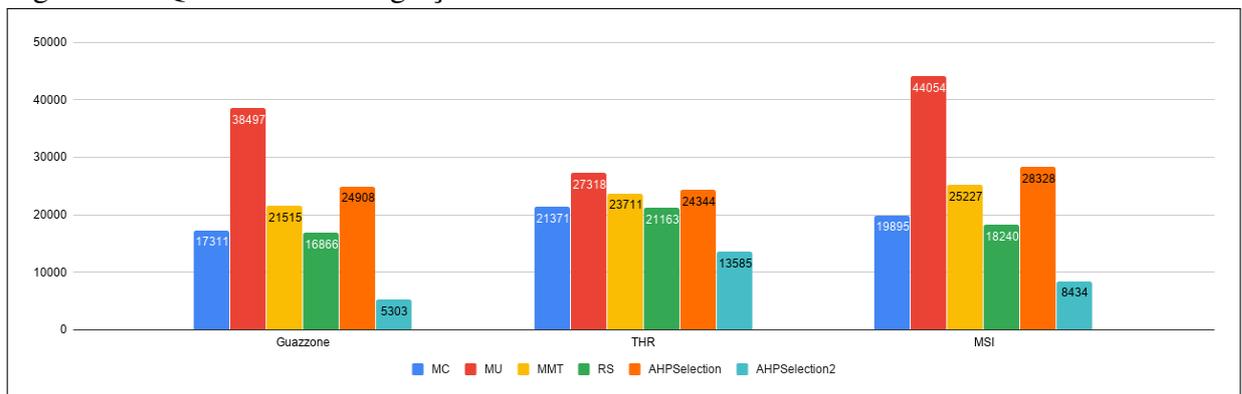
De acordo com a Figura 11 é possível observar que a política AHPSelection, em sua primeira versão, apesar de pouca diferença entre os consumos de energia, apresenta um comportamento de valor médio frente os valores das outras políticas. Por exemplo, no caso em que a AHPSelection atuou junto com a política de alocação Guazzone, a política proposta por esse trabalho apresentou um gasto energético de 109,03 Kwh em relação ao gasto médio de 109,9 Kwh pelas políticas concorrentes. Da mesma forma aconteceu quando a AHPSelection atuou com a política de alocação THR, onde gastou 169,81 Kwh diante de 165,81 Kwh de gasto médio pelas outras políticas, porém nesse caso apresentou um leve distanciamento devido a estrutura da AHP que equilibra tanto a energia como o desempenho e ainda por que a política

THR é a que mais consome energia comparado a Guazzone e MSI. Quando atuou com o MSI, onde gastou 93,13 Kwh em relação aos 94,11 Kwh de gasto médio pelas políticas adversárias. Esse comportamento era esperado, já que a AHP trabalha com os mesmos critérios que as outras políticas utilizam (nível de consumo de processador e memória). O aspecto positivo desses resultados é observar que, em nenhum dos casos, a política AHPSelection se destacou como a pior política, tendo um comportamento intermediário entre todos os traces. Isso acontece dado que, hierarquicamente, a AHP é estruturada a fim de analisar tanto o consumo de energia, baseado no critério de Utilização do Processador e o custo de migração, baseado na utilização da memória RAM.

Já na outra versão AHPSelection2, é nítido observar uma diminuição de energia muito significativa. Com a utilização da política de alocação THR, por exemplo, foi economizado 35,99 Kwh de energia em relação a segunda melhor política de seleção (RS). Assim, também aconteceu quando a AHPSelection trabalhou em conjunto com a política de alocação Guazzone, economizando 12,57 Kwh e com a política de alocação MSI, economizou 11,61 Kwh, ambos os resultados comparados com a segunda melhor colocada (MU), respectivamente. Tais resultados corroboram com as conclusões obtidas no trabalho de Matos *et al.* (2015), onde a espera pela decisão de migrar uma máquina virtual pode acarretar em economia de energia.

6.1.2 Quantidade de migrações

Figura 12 – Quantidade de migrações do trace PlanetLab



Fonte: Próprio autor

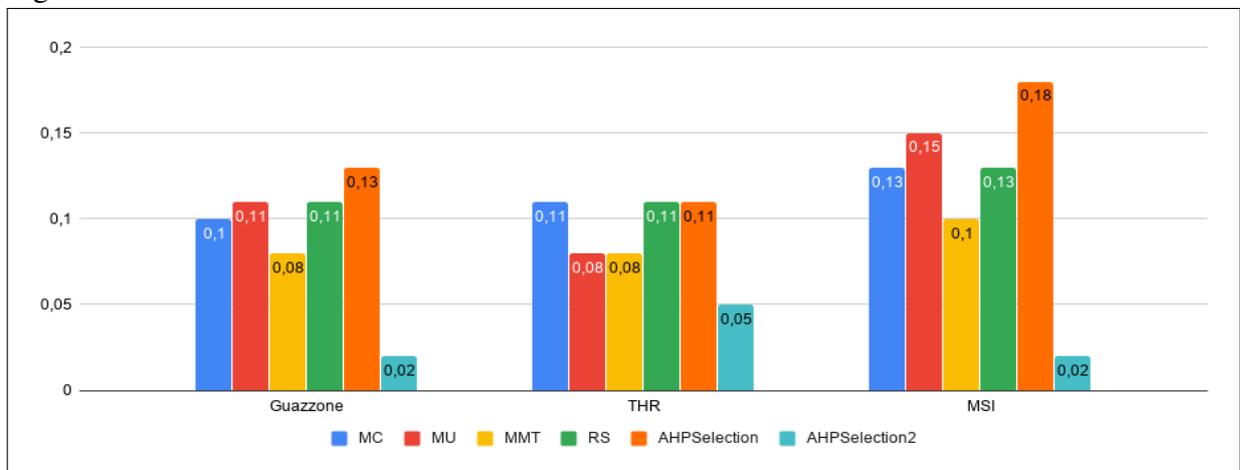
De acordo com a Figura 12 foi possível notar que a quantidade de migrações da AHPSelection, em sua primeira versão, manteve uma certa estabilidade em relação a quantidade de migrações de outras políticas. Analogamente ao consumo de energia, a AHPSelection, em

sua primeira versão, não se destacou como a política que mais fez migrações. É possível notar na Figura 12, que a AHPSelection, quando atuou com a política de alocação Guazzone, gastou, aproximadamente, somente 6% a mais que o número médio de migrações das outras políticas. De forma semelhante aconteceu com o THR e a MSI, onde a AHPSelection gastou, aproximadamente, 4% a mais do gasto médio com migrações pelas outras políticas e 5% a mais que as migrações médias das políticas concorrentes, respectivamente. Isso demonstra um forte indicativo da eficiência da AHPSelection em relação as questões de desempenho.

A AHPSelection2, segunda versão da AHPSelection, reduziu bastante a quantidade de migrações, em todos os casos. A política em questão, em média, economizou aproximadamente 15490 migrações em relação a média de migrações causadas pelas políticas concorrentes (MC, MU, MMT e RS). Tais valores são consequências da proposta de esperar um turno de sobrecarga para decidir qual Máquina Virtual migrar.

6.1.3 PDM

Figura 13 – PDM do trace PlanetLab



Fonte: Próprio autor

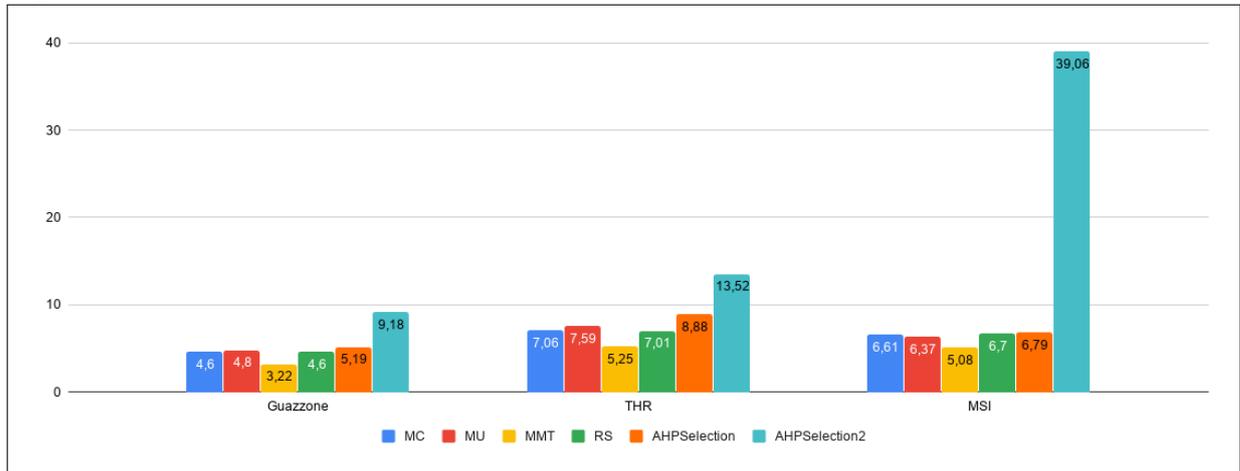
De acordo com a Figura 13, a AHPSelection, em sua primeira versão, apresentou resultados satisfatórios. Apesar de não ter obtido os menores resultados quando comparados com as demais políticas, o desempenho da mesma não se distanciou tanto da média das concorrentes (aproximadamente 0.03% em todos os três cenários). Este comportamento já era esperado, uma vez que a AHP tem como principal objetivo, reduzir o consumo de energia, em detrimento de algumas violações de SLA. Isso pode ser explicado pela proporcionalidade com a quantidade de migrações por política (Figura 12). Quanto maior o número de migrações, maior será a

quantidade de recursos dedicados a migração e, conseqüentemente, maiores serão os valores da métrica em questão.

Já analisando os valores obtidos pela AHPSelection2, é possível notar que a mesma conseguiu reduzir significativamente os valores da versão 1 da política: 80% no cenário com a Política de Alocação Guazzone, 50% com a Política de Alocação THR e 90% com a Política de Alocação MSI. Essa redução também é motivada pela proposta de somente realizar a migração quando o servidor estiver sobrecarregado a, pelo menos, um período. Note que os valores obtidos também acompanham a quantidade reduzida de migrações.

6.1.4 SLATAH

Figura 14 – SLATAH do trace PlanetLab



Fonte: Próprio autor

Os resultados obtidos na métrica SLATAH podem ser visualizados na Figura 14. Para ambas as versões da Política proposta neste trabalho, essa métrica foi aquela que ela obteve os piores resultados, a versão 1 bem menos do que a 2. Enquanto a versão 1 apresentou valores acima de 0.88%, 2.88% e 0.6% das médias das demais políticas concorrentes por cenário, a versão 2 mostrou um crescimento bastante elevado, principalmente quando analisado o cenário em conjunto com a Política de Alocação MSI.

De acordo com a Figura 14, a AHPSelection, em sua primeira versão, apesar de apresentar valores SLATAH mais altos que as demais, não comprometeu muito o desempenho geral do sistema, pois a quantidade de violações foram próximas entre as políticas. Esse aumento na SLATAH pode ser explicado devido ao comportamento que o *trace PlanetLab* assume. Nesse *trace* as aplicações consomem muitos recursos computacionais e/ou essas aplicações tem uma

alta variabilidade de consumo, ou seja, pode ocorrer que um servidor passe de um estado de baixo consumo para um estado de alto consumo rapidamente. Devido a isso, os servidores podem ficar sobrecarregados mais tempo ou de maneira mais rápida, o que contribui para a violação no SLATAH.

Já se avaliados os resultados do SLATAH da AHPSelection2, nota-se um elevado crescimento quando comparado com a versão 1 da mesma Política. Esse crescimento já era esperado, uma vez que a proposta de adiar a migração de máquinas virtuais, pode acarretar situações em que servidores com 100% de consumo de CPU, fiquem mais tempo nessa condição, o que incrementa o valor da SLATAH.

No caso específico da MSI, por exemplo, houve uma alta porcentagem de violação do SLATAH. Isso ocorre devido a alta consolidação que a Política de Alocação MSI causa, já que Filho (2018) alcança uma alta economia de energia no *trace PlanetLab* em seu trabalho. Dado que o AHPSelection2 utiliza o método de espera de um período pra economizar energia, pode ocorrer que servidores sobrecarregados fiquem nesse estado por mais tempo. A combinação desta elevada consolidação da MSIAllocator com o modo de espera de um período da AHPSelection2 explicam o resultado apresentado no gráfico.

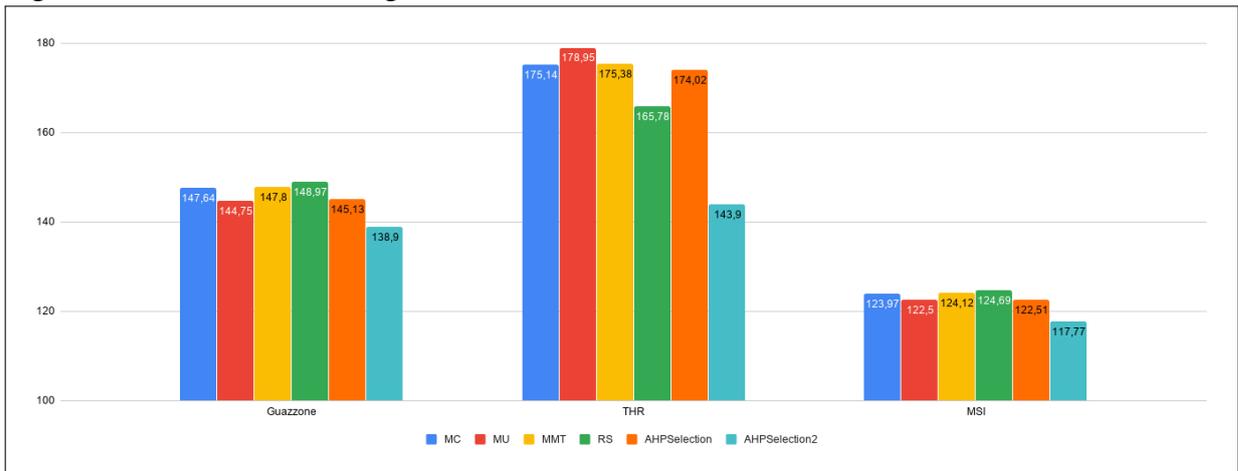
6.2 *Trace BitBrains*

Nesta seção serão apresentados os resultados dos desempenho das políticas de seleção com cada política de alocação utilizando o *trace Bitbrains*. Em cada subseção serão abordados os resultados de acordo com cada métrica já estabelecidas na Seção 5.2.

6.2.1 *Consumo de energia*

De acordo com a Figura 15 é possível notar que a política AHPSelection, em sua primeira versão, apresentou um bom comportamento frente ao valor médio das outras políticas. Quando a AHPSelection associou-se a política de alocação Guazzone, foi consumido 138,9 Kwh pela política proposta frente a 147,29 Kwh de consumo médio das políticas concorrentes. Da mesma maneira, ocorreu com a THR e MSI, com 143,9 Kwh contra 171,81 Kwh das concorrentes e 117,77 Kwh ante 123,82 Kwh das políticas oponentes, respectivamente. Logo, é notável o comportamento médio de gasto energético da AHPSelection. Observa-se que, de uma maneira geral, os resultados obtidos neste cenário são parecidos aqueles apresentados na Figura 11. Para

Figura 15 – Consumo de energia do trace BitBrains

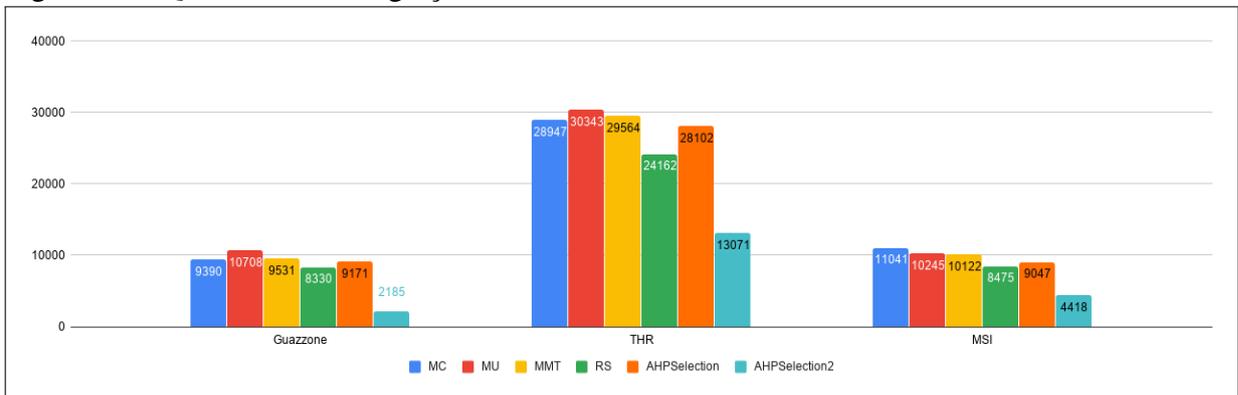


Fonte: Próprio autor

não ser tão repetitivo, sugere-se a leitura da Subseção 6.1.1 para maiores explicações sobre as razões desse comportamento.

6.2.2 Quantidade de migrações

Figura 16 – Quantidade de migrações do trace BitBrains



Fonte: Próprio autor

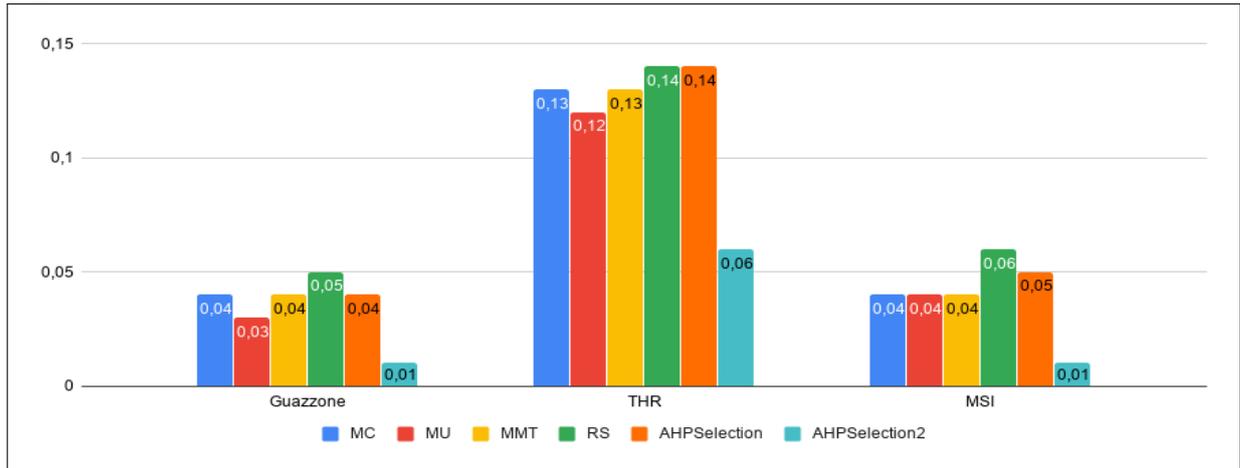
De acordo com a Figura 16 é possível notar que a quantidade de migrações da AHPSelection, em sua primeira versão, manteve uma certa estabilidade em relação a quantidade de migrações de outras políticas. Analogamente ao consumo de energia, a AHPSelection, em sua primeira versão, não se destacou como a política que mais fez migrações.

A AHPSelection2 reduziu drasticamente o número de migrações, em todos os cenários. A AHPSelection2 juntamente com a política de alocação Guazzone economizou 6145 migrações em relação a segunda colocada (RS). De maneira análoga, com a política de alocação

THR e MSI, economizou-se 11091 e 4057 migrações, respectivamente. Este comportamento foi o mesmo observado quando utilizado o trace do PlanetLab. Para maiores explicações sugere-se a leitura da Subseção 6.1.2.

6.2.3 PDM

Figura 17 – PDM do trace BitBrains



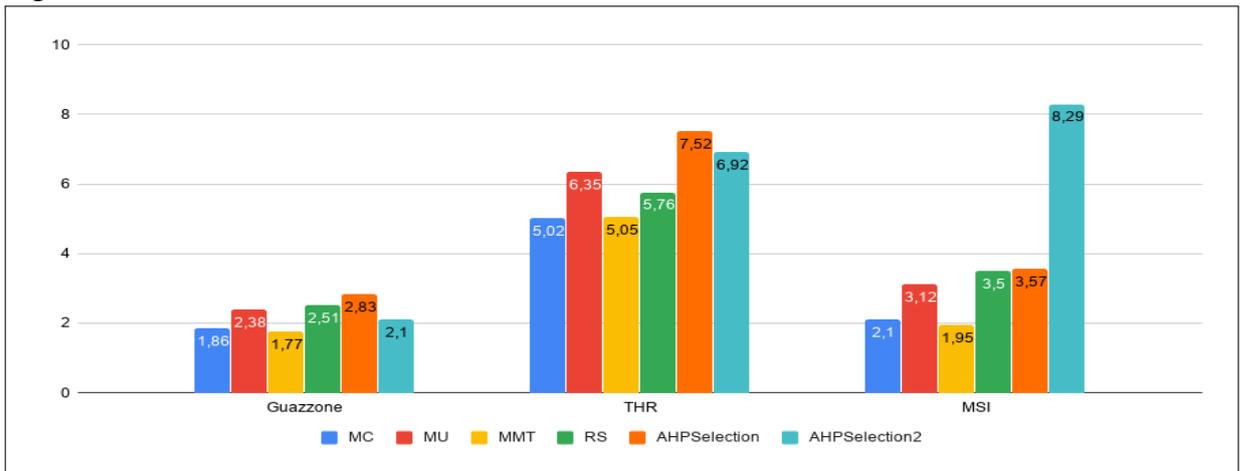
Fonte: Próprio autor

De acordo com a Figura 17 a AHPSelection, em sua primeira versão, apesar de não ser a menor violação entre algumas outras políticas, apresentou resultados satisfatórios dado que em nenhum dos casos a política AHPSelection apresentou-se como a pior política no caso da métrica de PDM. Já a AHPSelection2 reduziu drasticamente a violação de PDM, já que, em alguns casos, a política se negou a realizar a migração de uma Máquina Virtual e isso acaba influenciando nos resultados dessa métrica (Subseção 6.1.3). Por exemplo, no cenário com a Política Guazzone, a AHPSelection2 reduziu em 75% o número de PDM em relação a média das concorrentes. Da mesma forma ocorreu com a THR e MSI, com 53% e 78%, respectivamente.

6.2.4 SLATAH

De acordo com a Figura 18, a AHPSelection, em sua primeira versão, apesar de apresentar valores de SLATAH mais altos que as outras, não comprometeu muito o desempenho pois sua violação foi bem próximas das outras políticas. Por exemplo, a violação causada pela AHPSelection quando executou juntamente com a política de alocação Guazzone foi de 2,83% ante 2,13% da média de violações das outras políticas. Semelhantemente, aconteceu com a THR

Figura 18 – SLATAH do trace BitBrains



Fonte: Próprio autor

e a MSI.

Já quando analisada a AHPSelection2, observa-se um comportamento um pouco diferente daquele observado no cenário da seção anterior com o PlanetLab (Subseção 6.1.4). Neste último, o baixo número de migrações ocasionou um alto número de violações da métrica SLATAH em todas as políticas de alocação. Contudo, neste caso, observa-se uma redução significativa nos resultados encontrados, em alguns casos, obtendo valores abaixo do que algumas das políticas de seleção concorrentes. Acredita-se que este comportamento é impulsionado pela observação de que no *trace BitBrains* há menor ocorrência de servidores sobrecarregados em 100% comparado ao que ocorre no *trace PlanetLab*. Essa colocação foi motivada pelo fato de que a AHPSelection2, mesmo sendo a política que utilizou um período de espera e que afeta diretamente a métrica, seu valor de SLATAH ainda se apresentou menor quando comparada a algumas políticas, no caso do *trace BitBrains*. No *trace PlanetLab*, nesse caso, se apresentou como pior política em relação as violações de SLATAH, o que reforça a ideia de que o *trace* possui servidores que gastam mais tempo sobrecarregados quando comparado ao *trace BitBrains*. Note que, embora tenha conseguido reduzir bastante, o valor da SLATAH junto com a MSIAAllocator ainda está relativamente alto. Acredita-se que tal comportamento se deve ao alto índice de consolidação que a MSIAAllocator proporciona. Para melhores explicações, sugere-se a leitura da Subseção 6.1.4.

É importante destacar a importância desse resultado para a AHPSelection2, porque ele demonstra que, para alguns tipos de aplicações, a proposta de adiar a decisão de migrar uma máquina virtual pode ser benéfica para o datacenter. Se o leitor observar os gráficos obtidos com os resultados deste *trace*, percebe-se que a referida política conseguiu economizar energia e

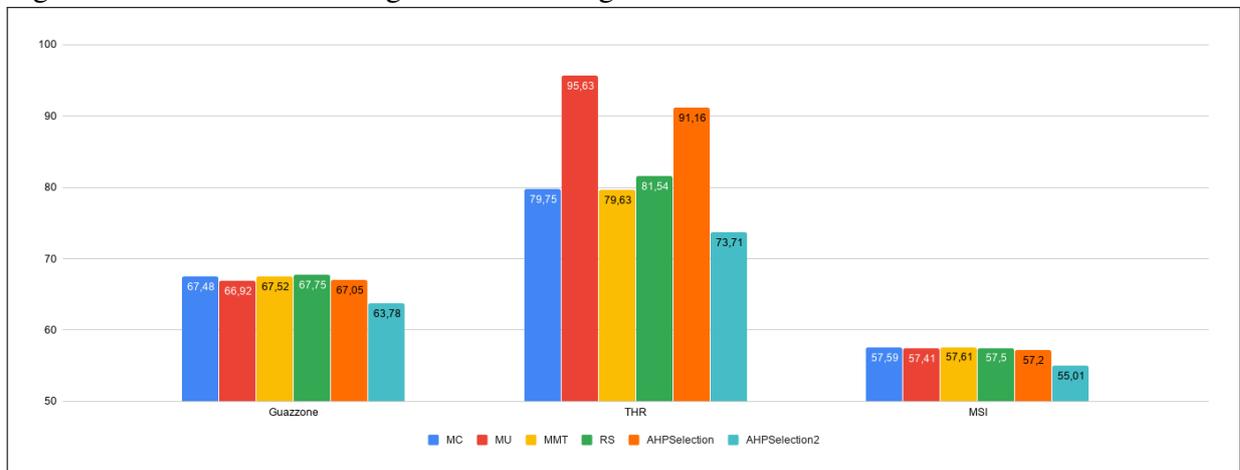
ainda obteve resultados intermediários as demais políticas.

6.3 Trace GoogleCluster

Nesta seção será apresentado os resultados dos desempenho das políticas de seleção com cada política de alocação utilizando o *trace GoogleCluster*. Em cada subseção será abordado os resultados de acordo com cada métrica já estabelecidas na Seção 5.2.

6.3.1 Consumo de energia

Figura 19 – Consumo de energia do trace GoogleCluster



Fonte: Próprio autor

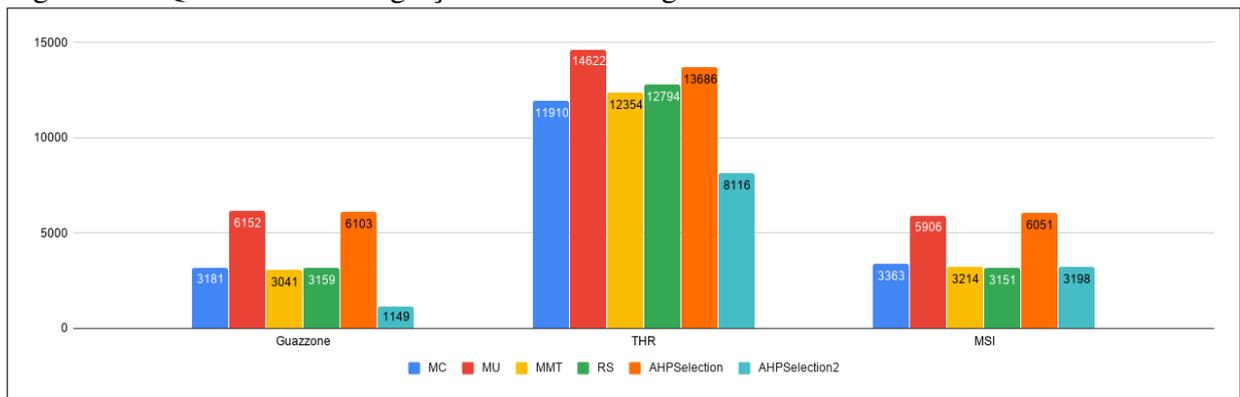
De acordo com a Figura 19 é possível notar que a política AHPSelection, em sua primeira versão, apesar de pouca diferença entre os consumos de energia, apresenta um bom comportamento de valor médio frente os outros valores das outras políticas. No caso, do Guazzone, por exemplo, a diferença entre o valor do AHPSelection, em sua primeira versão, e a média das políticas concorrentes foi de 0,54%, apenas. Na THR, a diferença foi de 8% e na MSI, somente 0,57% de diferença. Tal comportamento já era esperado, uma vez que a AHP escolhe a melhor opção com base em todos os critérios que são definidos. Como nesse caso os critérios são relacionados a utilização de processador e memória e as outras políticas utilizam, em sua maioria, as mesmas métricas, é nítido que a tendência seria ocorrer um balanceamento. Mas, em nenhum dos casos, a política AHPSelection se destacou como a pior política.

Já na outra versão AHPSelection2, é nítido observar uma diminuição de energia. Com a utilização da política de alocação THR, por exemplo, foi economizado 5,92 Kwh de energia

em relação a segunda melhor política de seleção (MMT). Assim, também aconteceu quando a AHPSelection trabalhou em conjunto com a política de alocação Guazzone, economizando 3,63 Kwh e com a política de alocação MSI, economizou 2,4 Kwh, ambos os resultados comparados com a segunda melhor colocada (MU), respectivamente.

6.3.2 Quantidade de migrações

Figura 20 – Quantidade de migrações do trace GoogleCluser



Fonte: Próprio autor

De acordo com a Figura 20 foi possível notar que a quantidade de migrações da AHPSelection, em sua primeira versão, manteve uma certa estabilidade em relação a quantidade de migrações de outras políticas. Apesar de não ser a menor em comparação com outras políticas, a AHPSelection teve um comportamento médio esperado, não extrapolando muitos os números apresentados pelas outras políticas e atingindo uma faixa de valores aceitável.

A AHPSelection2, segunda versão da política AHPSelection, reduziu drasticamente o número de migrações. De acordo com Matos *et al.* (2015) isso é causado pelo fato de que, algumas vezes, a política se recusa a escolher uma Máquina Virtual para migrar. Note que em todos os casos, a quantidade de migrações da AHPSelection2 foi a menor.

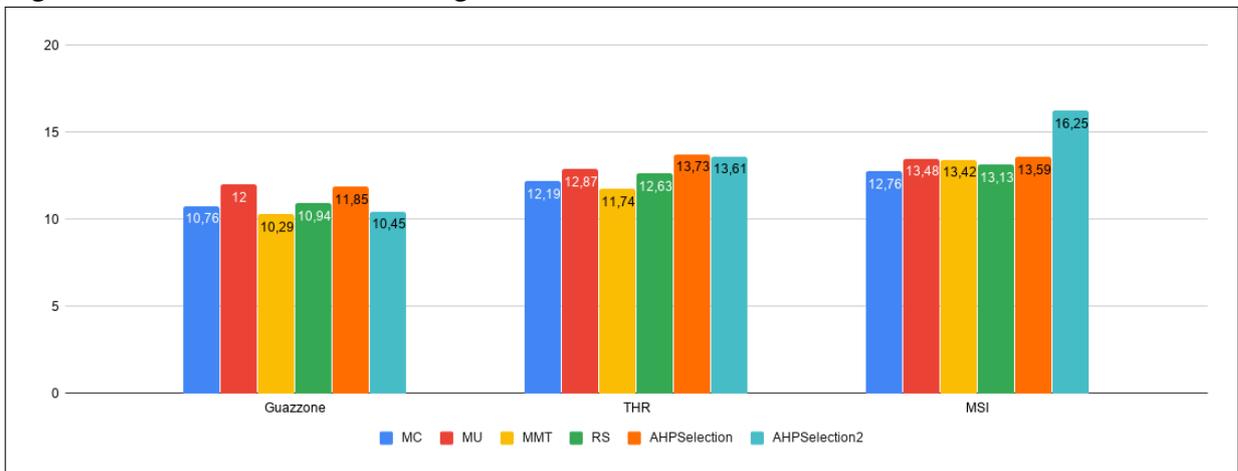
A AHPSelection2 juntamente com a política de alocação Guazzone economizou 1892 migrações em relação a segunda colocada. De maneira análoga, com a política de alocação THR economizou-se 3794 migrações. No MSI, a política AHPSelection2 somente perdeu para a RS. Mas, apesar da política AHPSelection2 apresentar quantidade de migrações maior que a RS, a diferença entre elas foi de somente 47 migrações, o que ainda torna os resultados da AHPSelection2 favoráveis.

6.3.3 PDM

Assim como aconteceu com Filho (2018), os resultados desta métrica com este *trace* obtiveram valores muito próximos a zero com quaisquer Política de Alocação (Guazzone, THR, MSI). Esse fenômeno ocorreu por que a quantidade de migrações que foram realizadas na simulação no *trace GoogleCluster* não foram suficientes para que a porcentagem de violação fosse significativa. Ressalta-se então que, a AHPSelection, tanto na primeira como na segunda versão, apresentou, assim como as outras, baixíssimas violações na métrica PDM.

6.3.4 SLATAH

Figura 21 – SLATAH do trace GoogleCluster



Fonte: Próprio autor

De acordo com a Figura 21 a AHPSelection, em sua primeira versão, apresentou resultados satisfatórios. Apesar de que, em alguns casos, não foi a melhor, ainda assim, não comprometeu muito o desempenho pois sua violação foi bem próximas das outras políticas.

Já os resultados da AHPSelection2, acompanharam a diminuição observada nos resultados do *trace* do BitBrains. Destaca-se a aproximação dos resultados obtidos em conjunto com a política MSIALlocator. Foi observado que cada *trace* apresenta um comportamento de sobrecarga diferente, onde o *trace PlanetLab* tem maior incidência de sobrecarga de 100% que o *trace BitBrains* e *GoogleCluster*. Estes resultados reforçam a ideia defendida na Subseção 6.2.4, onde, em cenários onde as aplicações não consomem muitos recursos ou variam muito suas requisições, a espera pela decisão pode ser a melhor solução, dado que o AHPSelection2 utiliza o método de espera de um período pra economizar energia. Dado a observância do

comportamento do *trace PlanetLab*, isso explica o por que do SLATAH ter se apresentado menor quando comparado a algumas políticas nos *traces BitBrains* e *GoogleCluster*.

7 CONCLUSÃO

Este trabalho definiu uma Política de Seleção de Máquinas Virtuais, denominada AHPSelection, que teve o objetivo de reduzir o consumo de energia em um ambiente de *data center*. A AHPSelection utilizou o método muito popular de tomada de decisão, o AHP, afim de escolher qual Máquina Virtual será utilizada para migração. A contribuição desse trabalho foi a proposta de uma nova política de seleção de Máquinas Virtuais utilizando um método já consolidado em processos de tomada de decisão e confrontar com algumas políticas já existentes. Além disso, o trabalho forneceu resultados quando a AHPSelection trabalha em conjunto com três diferentes políticas de alocação e utilizando um cenário simulado de um ambiente real.

Outra contribuição foi estabelecer um certo equilíbrio nos valores quando usada a primeira versão da AHPSelection, pois a AHP utilizada, em sua estrutura, pondera tanto a energia como o desempenho. Além disso, também foi proposta uma segunda versão da AHPSelection, chamada de AHPSelection2, utilizando uma abordagem de espera de um período de sobrecarga a fim de diminuir o consumo energético. Observou-se que esta nova versão apresentou bons resultados em cenários onde a pouca variação na demanda de recursos por parte das Máquinas Virtuais. Nos demais cenários, os resultados foram, ao menos promissores, cabendo alguns estudos mais aprofundados para aplicar a política no referido contexto.

Os resultados mostraram que as duas versões da AHPSelection se comportaram satisfatoriamente. Apesar da primeira versão não apresentar uma economia muito grande de energia, em alguns casos, a política proposta nesse trabalho foi melhor que algumas das políticas concorrentes e em nenhum dos casos a AHPSelection foi considerada a pior política. A segunda versão da AHPSelection apresentou uma grande economia de energia. Nas métricas de quantidade de migrações, PDM e SLATH as duas versões se comportaram de maneira satisfatória: a AHPSelection manteve uma estabilidade nas violações e a AHPSelection2 conseguiu obter bons resultados em algumas dessas métricas. Embora alguns dos resultados tenham se mostrados completamente desfavoráveis a AHPSelection2, a redução progressiva dessas adversidades chegando, em alguns cenários, a resultados melhores do que aqueles observados em outras políticas, encorajam, pelo menos, um estudo mais profundo do impacto do uso da AHPSelection2 em outros cenários.

Como trabalhos futuros, poderiam ser feitos novos testes em outros *traces* e em outros cenários e comparar seu comportamento com os resultados oferecidos por esse trabalho. Além disso, seria importante estudar a viabilidade da AHP economizar mais energia inserindo

novos critérios. Outro estudo a se fazer é o impacto da política AHPSelection nas finanças do Provedor de Serviço, ou seja, se a economia de energia realizada compensa as possíveis multas que podem ocorrer devido as violações de SLA.

REFERÊNCIAS

- ALJABRE, A. Cloud computing for increased business value. In: . [S.l.: s.n.], 2012.
- AOUAR, S. **Estrutura da virtualização tradicional**. 2018. Disponível em: <<https://www.salimaouar.com.br/artigo/entenda-o-que-e-o-lxc-linux-containers>>. Acesso em: 21 mai. 2019.
- BAJWA, G.; CHOO, E. U.; WEDLEY, W. C. Effectiveness analysis of deriving priority vectors from reciprocal pairwise comparison matrices. **Asia-Pacific Journal of Operational Research**, v. 25, n. 03, p. 279–299, 2008. Disponível em: <<https://doi.org/10.1142/S0217595908001754>>.
- Barroso, L. A.; Hölzle, U. The case for energy-proportional computing. **Computer**, v. 40, n. 12, p. 33–37, Dec 2007. ISSN 0018-9162.
- BELOGLAZOV, A.; BUYYA, R. Adaptive threshold-based approach for energy-efficient consolidation of virtual machines in cloud data centers. In: **Proceedings of the 8th International Workshop on Middleware for Grids, Clouds and e-Science**. New York, NY, USA: ACM, 2010. (MGC '10), p. 4:1–4:6. ISBN 978-1-4503-0453-5. Disponível em: <<http://doi.acm.org/10.1145/1890799.1890803>>.
- BELOGLAZOV, A.; BUYYA, R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. **Concurrency and Computation: Practice and Experience**, v. 24, 09 2012.
- BELOGLAZOV, A.; BUYYA, R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. **Concurrency and Computation: Practice and Experience**, v. 24, 09 2012.
- BELOGLAZOV, A.; BUYYA, R. Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers. **Concurr. Comput. : Pract. Exper.**, John Wiley and Sons Ltd., Chichester, UK, v. 24, n. 13, p. 1397–1420, set. 2012. ISSN 1532-0626. Disponível em: <<http://dx.doi.org/10.1002/cpe.1867>>.
- BOCKEN, N. M.; MUGGE, R.; BOM, C. A.; LEMSTRA, H.-J. Pay-per-use business models as a driver for sustainable consumption: Evidence from the case of homie. **Journal of Cleaner Production**, v. 198, p. 498 – 510, 2018. ISSN 0959-6526. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0959652618320109>>.
- CALHEIROS, R. N.; RANJAN, R.; BELOGLAZOV, A.; ROSE, C. A. F. D.; BUYYA, R. Cloudsim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. **Softw. Pract. Exper.**, John Wiley & Sons, Inc., New York, NY, USA, v. 41, n. 1, p. 23–50, jan. 2011. ISSN 0038-0644. Disponível em: <<http://dx.doi.org/10.1002/spe.995>>.
- Cao, Z.; Dong, S. Dynamic vm consolidation for energy-aware and sla violation reduction in cloud computing. In: **2012 13th International Conference on Parallel and Distributed Computing, Applications and Technologies**. [S.l.: s.n.], 2012. p. 363–369. ISSN 2379-5352.

CARLI, P.; DELAMARO, M.; SALOMON, V. Identification and prioritization of critical success factors in a digital factory implementation. **Produção**, v. 20, 12 2010.

CASTELLS, M. **A Galáxia Internet: reflexões sobre a Internet, negócios e a sociedade**. Jorge Zahar, 2003. ISBN 9788571107403. Disponível em: <<https://books.google.com.br/books?id=nCKFFmWOnNYC>>.

CLARK, C.; FRASER, K.; HAND, S.; HANSEN, J. G.; JUL, E.; LIMPACH, C.; PRATT, I.; WARFIELD, A. Live migration of virtual machines. In: **Proceedings of the 2Nd Conference on Symposium on Networked Systems Design & Implementation - Volume 2**. Berkeley, CA, USA: USENIX Association, 2005. (NSDI'05), p. 273–286. Disponível em: <<http://dl.acm.org/citation.cfm?id=1251203.1251223>>.

CLARK, D. **Google discloses carbon footprint for the first time**. 2011. Disponível em: <<https://www.theguardian.com/environment/2011/sep/08/google-carbon-footprint>>. Acesso em: 23 mai. 2019.

COELHO, F. de A.; CALZAVARA, G. S.; LUCIA, R. di. **Virtualização - VMWare e Xen**. 2008. Disponível em: <https://www.gta.ufrj.br/grad/09_1/versao-final/virtualizacao/conceito/%20de%20virtualizacao.html>. Acesso em: 23 mai. 2019.

COSTA, D. **JAVA EM REDE**. BRASPORT, 2008. ISBN 9788574523361. Disponível em: <https://books.google.com.br/books?id=_Fr0ooiSIYUC>.

CRAWFORD, G.; WILLIAMS, C. A note on the analysis of subjective judgment matrices. **Journal of Mathematical Psychology**, v. 29, n. 4, p. 387 – 405, 1985. ISSN 0022-2496. Disponível em: <<http://www.sciencedirect.com/science/article/pii/0022249685900021>>.

DANILAK, R. **Why Energy Is A Big And Rapidly Growing Problem For Data Centers**. 2017. Disponível em: <<https://www.forbes.com/sites/forbestechcouncil/2017/12/15/why-energy-is-a-big-and-rapidly-growing-problem-for-data-centers/#23dad7135a30>>. Acesso em: 22 mai. 2019.

DATA FLAIR. **Cloud Computing Technology (CCT)**. 2019. Disponível em: <<https://data-flair.training/blogs/cloud-computing-technology/>>. Acesso em: 14 mai. 2019.

de Matos, F. F. S. B.; Celestino, J.; Cardoso, A. R. Vbalance: A selection policy of virtual machines for load balancing in cloud computing. In: **2015 IEEE Symposium on Computers and Communication (ISCC)**. [S.l.: s.n.], 2015. p. 770–775.

FAN, X.; WEBER, W. dietrich; BARROSO, L. A. Power provisioning for a warehouse-sized computer. In: **In Proceedings of ISCA**. [S.l.: s.n.], 2007.

FILHO, F. M. V. Msiallocator: aplicação da meta-heurística ils para o problema de alocação de máquinas virtuais em uma data center. 2018. Disponível em: <<http://www.repositorio.ufc.br/handle/riufc/38757>>.

FILHO, N. M.; PEREIRA, M. V. F. Programação linear. In: . [S.l.: s.n.], 1980.

Gleb B. **Cloud Services (SaaS, PaaS, IaaS)**. 2017. Disponível em: <<https://rubygarage.org/blog/iaas-vs-paas-vs-saas>>. Acesso em: 21 mai. 2019.

GUAZZONE, M.; ANGLANO, C.; CANONICO, M. Exploiting vm migration for the automated power and performance management of green cloud computing systems. In: **Proceedings of the First International Conference on Energy Efficient Data Centers**. Berlin, Heidelberg: Springer-Verlag, 2012. (E2DC'12), p. 81–92. ISBN 978-3-642-33644-7. Disponível em: <http://dx.doi.org/10.1007/978-3-642-33645-4_8>.

HO, W. Integrated analytic hierarchy process and its applications – a literature review. **European Journal of Operational Research**, v. 186, n. 1, p. 211 – 228, 2008. ISSN 0377-2217. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221707000872>>.

Hussein, S. R.; Alkabani, Y.; Mohamed, H. K. Green cloud computing: Datacenters power management policies and algorithms. In: **2014 9th International Conference on Computer Engineering Systems (ICCES)**. [S.l.: s.n.], 2014. p. 421–426.

IBGE. **Domicílios particulares permanentes**. 2010. Disponível em: <<https://cidades.ibge.gov.br/brasil/ce/fortaleza/pesquisa/23/47427?tipo=ranking&indicador=47428>>. Acesso em: 22 mai. 2019.

KAHLIG, J. **Linear Programming Problems**. 2019. Disponível em: <<https://www.math.tamu.edu/~joe.kahlig/notes/141/ch03-2.pdf>>. Acesso em: 18 dez. 2019.

KOOMEY, J.; TURNER, P.; STANLEY, J.; TAYLOR, B. A simple model for determining true total cost of ownership for data centers. 01 2007.

MAHMOUD, Q. H. **Service-Oriented Architecture (SOA) and Web Services: The Road to Enterprise Application Integration (EAI)**. 2005. Disponível em: <<https://www.oracle.com/technetwork/articles/javase/soa-142870.html>>. Acesso em: 23 mai. 2019.

MANN, Z.; SZABÓ, M. Which is the best algorithm for virtual machine placement optimization? **Concurrency and Computation: Practice and Experience**, p. e4083, 03 2017.

MARZULLO, F. **SOA na Prática**. NOVATEC, 2009. ISBN 9788575222010. Disponível em: <<https://books.google.com.br/books?id=h5WjAwAAQBAJ>>.

Matos, F. F. S. B. de; Celestino, J.; Cardoso, A. R. Vbalance: A selection policy of virtual machines for load balancing in cloud computing. In: **2015 IEEE Symposium on Computers and Communication (ISCC)**. [S.l.: s.n.], 2015. p. 770–775.

MELL, P.; GRANCE, T. **The NIST Definition of Cloud Computing**. 2011. Disponível em: <<https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>>. Acesso em: 23 mai. 2019.

MOGHADDAM, F. A.; LAGO, P.; GROSSO, P. Energy-efficient networking solutions in cloud-based environments: A systematic literature review. **ACM Computing Surveys**, v. 47, p. 1–32, 05 2015.

MOGHADDAM, S. M.; PIRAGHAJ, S. F.; OSULLIVAN, M.; WALKER, C.; UNSWORTH, C. Energy-efficient and sla-aware virtual machine selection algorithm for dynamic resource allocation in cloud data centers. In: . [S.l.: s.n.], 2018. p. 103–113.

MONKS, T. Operational research as implementation science: Definitions, challenges and research priorities. **Implementation Science**, v. 11, 05 2016.

NEVES, A. J. da S.; CAMANHO, R. The use of ahp for it project prioritization – a case study for oil gas company. **Procedia Computer Science**, v. 55, p. 1097 – 1105, 2015. ISSN 1877-0509. 3rd International Conference on Information Technology and Quantitative Management, ITQM 2015. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1877050915015513>>.

Niu, D.; Feng, C.; Li, B. A theory of cloud bandwidth pricing for video-on-demand providers. In: **2012 Proceedings IEEE INFOCOM**. [S.l.: s.n.], 2012. p. 711–719. ISSN 0743-166X.

NOROUZI-KHANGAH, B.; MOHAMMADSADEGHI-AZAD, M. B.; HOSEYNI, S. M.; HOSEYNI, S. M. Performance assessment of cooling systems in data centers; methodology and application of a new thermal metric. **Case Studies in Thermal Engineering**, v. 8, p. 152 – 163, 2016. ISSN 2214-157X. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2214157X1630034X>>.

PANDYA, S. S. Green cloud computing. 2014. ISSN 0974-2239.

PATEL, P.; RANABAHU, A.; SHETH, A. Service level agreement in cloud computing. In: . [S.l.: s.n.], 2009.

PENSAMENTO VERDE. **Você sabe o que é TI Verde?** 2013. Disponível em: <<https://www.pensamentoverde.com.br/sustentabilidade/voce-sabe-o-que-e-ti-verde/>>. Acesso em: 23 mai. 2019.

PONEMON INSTITUTE. **Cost to Support Compute Capacity**. 2016. Disponível em: <<https://www.vertiv.com/globalassets/documents/media-kits/afcom-dcw-fall-2016/reports/costtosupportcomputereport.pdf>>. Acesso em: 22 mai. 2019.

PONTES, V. **SOA e computação em nuvem**. 2010. Disponível em: <<https://viniuspontes.wordpress.com/2010/10/22/soa-e-computacao-em-nuvem/>>. Acesso em: 23 mai. 2019.

PORTO, M. A. G.; BANDEIRA, A. A. O processo decisório nas organizações. 2006.

PROCACCIANTI, G.; LAGO, P.; BEVINI, S. A systematic literature review on energy efficiency in cloud software architectures. **Sustainable Computing: Informatics and Systems**, v. 7, p. 2 – 10, 2015. ISSN 2210-5379. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S2210537914000900>>.

Qiu, C.; Shen, H.; Chen, L. Towards green cloud computing: Demand allocation and pricing policies for cloud service brokerage. **IEEE Transactions on Big Data**, p. 1–1, 2018. ISSN 2332-7790.

SAATY, T. The analytic hierarchy process: Decision making in complex environments. In: . [S.l.: s.n.], 1984.

SAATY, T. Decision making with the analytic hierarchy process. **Int. J. Services Sciences Int. J. Services Sciences**, v. 1, p. 83–98, 01 2008.

SANTOS, T. **Fundamentos da computação em nuvem**. Editora Senac São Paulo, 2018. (Universitária). ISBN 9788539611980. Disponível em: <<https://books.google.com.br/books?id=F4ZIDwAAQBAJ>>.

SEDGWICK, P. Pearson's correlation coefficient. **BMJ**, BMJ Publishing Group Ltd, v. 345, 2012. Disponível em: <<https://www.bmj.com/content/345/bmj.e4483>>.

SOHRABI, M. K.; GHODS, V.; FARD, Y. Z. A novel virtual machine selection policy for virtual machine consolidation. In: . [S.l.: s.n.], 2018. p. 28–32.

Teixeira, M. M.; Bestavros, A. End-to-end informed vm selection in compute clouds. In: **2015 IEEE International Conference on Communication Workshop (ICCW)**. [S.l.: s.n.], 2015. p. 1934–1939. ISSN 2164-7038.

VAQUERO, L.; RODERO-MERINO, L.; CACERES, J.; LINDNER, M. A. A break in the clouds: Towards a cloud definition. **Computer Communication Review**, v. 39, p. 50–55, 01 2009.

VARGAS, R. V.; PMP, I.-B. Using the analytic hierarchy process (ahp) to select and prioritize projects in a portfolio. **PMI Global Congress**, 01 2010.

VAUGHAN, A. **How viral cat videos are warming the planet**. 2015.

Disponível em: <<https://www.theguardian.com/environment/2015/sep/25/server-data-centre-emissions-air-travel-web-google-facebook-greenhouse-gas>>. Acesso em: 22 mai. 2019.

VERMA, A.; DASGUPTA, G.; NAYAK, T. K.; DE, P.; KOTHARI, R. Server workload analysis for power minimization using consolidation. In: **Proceedings of the 2009 Conference on USENIX Annual Technical Conference**. Berkeley, CA, USA: USENIX Association, 2009. (USENIX'09), p. 28–28. Disponível em: <<http://dl.acm.org/citation.cfm?id=1855807.1855835>>.

WESTPHALL, C.; VILLARREAL, S. Princípios e tendências em green cloud computing. **Revista Eletrônica de Sistemas de Informação**, v. 12, p. 7, 04 2013.