

Algoritmo de Escalonamento de MVs com Aprendizado de Máquina para Nuvens Computacionais com foco em Energia

Renato Tanaka, Kleiton Pereira, Gustavo B. da Silva,
Maurício A. Pillon, Guilherme P. Koslovski

¹ LabP2D - Universidade do Estado de Santa Catarina – UDESC

{renato.tanaka, kleiton.pereira, gustavo.silva}@edu.udesc.br,
{mauricio.pillon, guilherme.koslovski}@udesc.br

Resumo. A virtualização de recursos computacionais, sobretudo, através de Máquinas Virtuais (MVs), possibilitou, entre outros fatores, o surgimento de nuvens computacionais. Escalabilidade, flexibilidade e elasticidade são algumas das principais características de nuvens. Todavia, o fornecimento de recursos e qualidade de serviço em nuvens dependem de algoritmos de escalonamento de Máquina Virtual (MVs) online. O presente trabalho apresenta o ANJO, algoritmo de escalonamento de MVs para nuvem computacional. ANJO apoia-se em técnicas de aprendizado de máquina e *overbooking* escalonando MVs em nuvens *Infrastructure as a Service (IaaS)* buscando minimizar o custo energético.

1. Introdução

O fornecimento de recursos computacionais sob demanda, implementado através de plataformas de nuvens computacionais, tem movimentado pesquisas científicas e novos negócios na indústria nesta última década. Os modelos de serviços de nuvens (*Infrastructure as a Service (IaaS)*, *Platform as a Service (PaaS)* e *Software as a Service (SaaS)*) diferem no nível de abstração de acesso aos serviços fornecidos. O modelo de serviço com menor nível de abstração, IaaS, entrega a gerência integral de recursos virtuais (*e.g.*, vCPU, *switches*, armazenamento local ou efêmero) aos seus inquilinos.

Um *Cloud Operating System (COS)* aloca a Máquina Virtual (MV) solicitada pelo inquilino em um servidor físico de acordo com a política de alocação da nuvem, implementada através de algoritmos de escalonamento de MVs. Inquilinos não têm consciência da coexistência de MVs em um mesmo servidor físico e, os administradores da nuvem não controlam aplicações e sistema operacional interno a MV. Aos algoritmos de escalonamento de MVs, associados a COS, cabe a difícil tarefa de escolha de alocação do melhor servidor físico compatível com a solicitação do inquilino. A escolha deve estar de acordo com a política de distribuição de recursos da nuvem, porém não tem conhecimento prévio do uso efetivo de recursos da aplicação que executa dentro da MV.

Inquilinos e administradores estabelecem contratos de fornecimento de serviços, chamados de *Service Level Agreement (SLA)*. Inquilinos pagam pelo uso dos recursos por períodos (*pay-as-you-use*) ou pela alocação do recurso (*pay-as-you-go*), independente do uso. As políticas de alocação de recursos equalizam métricas como disponibilidade, consolidação e consumo de energia. As técnicas de consolidação de MVs e de *overbooking* são práticas comuns para a redução do consumo de energia. Consolidação de MV consiste na hospedagem de MVs no menor número de servidores físicos possível,

porém sem sobreposição de recursos virtuais. *Overbooking* caracteriza-se pela prática de alocação de MVs, em servidores físicos, com capacidade de recursos físicos inferior a soma dos recursos virtuais solicitados pelo conjunto de MVs hospedadas no servidor. Se por um lado, o uso destas técnicas favorece a redução de energia, por outro, prejudica a disponibilidade e qualidade dos serviços. Portanto, a intensidade de aplicação destas técnicas devem considerar o equilíbrio entre os fatores conflitantes citados anteriormente.

O aprendizado de máquina por reforço é uma técnica vinculada a grande área da Inteligência Artificial e utilizada há décadas na tomada de decisão. O problema de alocação de recursos em nuvens computacionais pode ser modelado através de várias métricas. Assim que o número de métricas consideradas aumenta, a modelagem pode tornar-se *NP Hard*. Os algoritmos de escalonamento operacionais e incorporados na maioria dos COS estão baseados em heurísticas ou soluções simplistas, tais como *First Come First Served* (FCFS), *Shortest Job First* (SJF) ou *Round Robin* (RR). Recentemente, o aprendizado de máquina passou a ser aplicado na tomada de decisão de alocação de MV em ambientes de nuvem computacional. Uma abordagem aplicada a esta área foi o modelo de aprendizado por reforço *Deep Q-Network*, constituído de uma rede neural, retroalimentada por um valor Q , recalculado a cada interação.

Neste contexto, o presente trabalho propõe o ANJO, algoritmo de escalonamento para nuvens computacionais apoiado no aprendizado de máquina com foco na redução do consumo de energia. O texto está organizado da seguinte forma: na Seção 2, são apresentados alguns algoritmos de alocação de recursos em nuvens computacionais e aprendizado por reforço; na Seção 3, encontra-se descrito o algoritmo de escalonamento ANJO; Finalizando, com as considerações parciais e trabalhos futuros (Seção 4).

2. Trabalhos Correlatos

Ambientes de nuvem computacional operacionais atualmente, apoiam-se em algoritmos de escalonamento clássicos, utilizados há décadas nas áreas de escalonamento de processos em sistemas operacionais ou em ambientes distribuídos, tais como grades computacionais. Trabalhos como [Meriam and Tabbane 2016] elaboraram análises detalhadas de algoritmos como, *First Come First Served* (FCFS), *Shortest Job First* (SJF) e *Round Robin* (RR). FCFS caracteriza-se pela aplicação da política simplista de escalonamento na ordem de chegada das requisições, evitando inanição e operando de forma justa, se observado o direito de chegada. O SJF prioriza processos, jobs ou MVs que solicitam o recurso por menos tempo. O segundo critério é o mesmo do FCFS, ordem de chegada. Este algoritmo não é imune a inanição. O RR é o primeiro a associar de forma diferenciada servidores em filas distintas, aplicando uma fila a cada servidor.

Os algoritmos clássicos têm tempo de execução compatíveis com problemas de escalonamento *online*, onde várias solicitações chegam simultaneamente, muitas vezes em forma de rajadas, e exigem respostas rápidas. Porém, pecam na análise detalhada de um maior conjunto de métricas, pois seus mecanismos não são escaláveis. A abordagem de aprendizado de máquina no contexto de escalonamento de recursos foi, primeiramente, aplicada por Gonhi e Gupta [Gondhi and Gupta 2017]. Os resultados apresentados neste trabalho são satisfatórios, considerando um maior número de métricas e obtendo tempo de execução aceitável para escalonamento de recursos *online*.

Recentemente, Cheng *et al* (DRL-Cloud) aplicou a técnica de aprendi-

zado de máquina por reforço diretamente em ambiente de nuvem computacional [Cheng et al. 2018]. Este trabalho inovou apoiando-se na abordagem de *Deep Q-Learning*, partindo do treinamento de uma rede neural, o algoritmo toma uma decisão considerada ótima. A política de escalonamento indicado a rede neural do DRL-Cloud considera o consumo de energia como essencial e visa minimizar o consumo total de energia do *Data Center* (DC). O aprendizado do DRL-Cloud está centrado na consolidação de MVs em ambientes de nuvens computacionais. No entanto, não explora a técnica de *overbooking*.

Ainda no contexto de nuvem computacional, porém com foco na caracterização das requisições de inquilinos, tem-se o trabalho de Caglar [Caglar and Gokhale 2014]. Este trabalho propõe a aplicação da técnica de otimização através do *overbooking* apoiando-se na técnica de aprendizado de máquina com redes neurais. Como subsídio a sua proposta, os autores caracterizaram os dados anônimos e públicos de requisições de MVs na nuvem do Google. Eles constataram que os usuários estudados tendem a superestimar suas requisições de recursos virtuais, por receio de perda de qualidade de serviço ou desconhecimento de suas necessidades. Portanto, se usuários costumam alocar mais recursos do que efetivamente utilizam, o cenário de nuvem computacional é propício a aplicação da técnica de *overbooking*. Os resultados de simulação deste trabalho mostram que a técnica de *overbooking* pode reduzir o número de servidores físicos ativos sem desrespeitar a SLA.

O levantamento bibliográfico mostrou, através de uma pesquisa não exaustiva, a importância do estudo de algoritmos de escalonamento de recursos em nuvem computacional e a recente aplicação das técnicas de aprendizado de máquina e *overbooking*. Constatou-se também uma lacuna na pesquisa científica no que se refere a utilização conjunta de todas as técnicas estudadas no contexto de nuvem computacional.

3. ANJO: Algoritmo de Escalonamento de MVs em Nuvem Computacional

A concepção de um algoritmo de escalonamento para nuvem computacional é a principal contribuição deste trabalho. Este algoritmo baseia-se no aprendizado de máquina por reforço com foco na redução do consumo de energia. O algoritmo é descrito através do uso dos componentes do *OpenAI gym*¹. O algoritmo é encapsulado em uma rede neural que se auto alimenta a cada interação através do cálculo de uma recompensa.

O modelo de entrada do algoritmo considera a existência de N servidores físicos constituídos de R recursos ($\langle vCPU, RAM \rangle$). Os servidores possuem recursos físicos com capacidades pré-definidas e fixas. A unidade mínima de alocação é a MV, cujo solicitação descreve somente os recursos R necessários. A rede neural recebe uma fila de solicitações de alocação de MVs e produz, como saída, a determinação $MV \rightarrow$ servidor físico. O sucesso do algoritmo depende do cálculo da recompensa e, por sua vez, o cálculo da recompensa das métricas e pesos atribuídos aos componentes do cálculo. Por convenção, uma recompensa deve retornar um valor entre 0 e 1. Com base nos trabalhos correlatos, considerou-se no cálculo da recompensa, a fragmentação dos recursos, o *overbooking* e a consolidação de MV (descrita na equação 1).

$$reward = 1 - \left(media \left(\frac{taxa_{overbookingCPU}}{limiar_{overbooking}}, \frac{taxa_{overbookingRAM}}{limiar_{overbooking}} \right) + frag \right) \quad (1)$$

¹<https://github.com/openai/gym>

Os resultados preliminares sobre a acurácia da recompensa foram insatisfatórios, demonstrando, na fase de treinamento, que a complexidade das métricas da recompensa afetam o aprendizado da rede neural. A nova estratégia foi a simplificação do cálculo da recompensa, apoiando-se somente na taxa de *overbooking*. Os recursos do conjunto *R* considerados foram *CPU* e *RAM*, delimitados por um limiar e taxa de uso. A fórmula de cálculo da recompensa simplificada está descrita na Equação 2.

$$reward = 1 - \max\left(\frac{taxa_{overbookingCPU}}{limiar_{overbooking}}, \frac{taxa_{overbookingRAM}}{limiar_{overbooking}}\right) \quad (2)$$

O método de análise dos resultados do algoritmo ANJO escolhido foi a simulação. O código fonte do Simulador ANJO ainda não disponível publicamente, pois encontra-se em fase de testes. Ele foi desenvolvido na linguagem de programação Python versão 3.6.5 e bibliotecas de Aprendizado de Máquina Keras (versão 2.2.4) e Tensorflow (versão 1.12.0).

4. Considerações Parciais & Trabalhos Futuros

A preocupação com o consumo de energia de DC em ambiente de nuvens computacionais é uma realidade. Algoritmos de escalonamento de COS são responsáveis pela alocação de MVs e suas escolhas têm impactos direto no consumo de energia. O uso de técnicas de *overbooking*, consolidação de MV e tomada de decisão com aprendizado de máquina são alternativas para possibilitar o consumo consciente de energia em nuvens computacionais. Finalmente, este trabalho apresenta um estudo preliminar do ANJO, um algoritmo de escalonamento de MV em nuvem computacionais direcionado por recompensa para redes neurais. Como trabalhos futuros, a curto prazo, o ANJO carece de maior detalhamento do modelo e, sobretudo, uma análise comparativa dos resultados das diferentes equações de recompensa. Além disso, o Simulador ANJO deve ter apurado e disponibilizado publicamente para uso da comunidade.

Referências

- Caglar, F. and Gokhale, A. (2014). iOverbook: Intelligent Resource-Overbooking to Support Soft Real-Time Applications in the Cloud. In *2014 IEEE 7th International Conference on Cloud Computing*, pages 538–545.
- Cheng, M., Li, J., and Nazarian, S. (2018). DRL-Cloud: Deep reinforcement learning-based resource provisioning and task scheduling for cloud service providers. In *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 129–134.
- Gondhi, N. K. and Gupta, A. (2017). Survey on machine learning based scheduling in cloud computing. In *Proceedings of the 2017 International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence, ISMSI '17*, pages 57–61, New York, NY, USA. ACM.
- Meriam, E. and Tabbane, N. (2016). A Survey on Cloud Computing Scheduling Algorithms. In *2016 Global Summit on Computer Information Technology (GSCIT)*, pages 42–47.