ulm university universität ulm

Image source : https://medium.com/@dhiraj.p.rai/jaya-optimization-algorithm-16da8708691b
https://de.wikipedia.org/wiki/Datei:Python_logo_and_wordmark.svg

Bachelor/Master Thesis
Design and implement a framework for cloud optimization algorithms in Python

2019-04-25

Context

A data centre operator aims to maximise it’s resource utilisation while achieving this with minimum number of servers, energy and resources to increase profit. Users aim to optimise on Cloud application level by scaling up and down according to user demands. Another aspect of optimization can be seen in cross data centre solutions, where Cloud applications are distributed across several vendors in order to avoid vendor lock-in and increase reliability.

The placement and migration problem of VMs in Cloud data centre is illustrated as a framework based on mathematical optimisation and objective function minimisation. VM allocation issue is considered as an NP-complete problem as we need to find out combinatorial optimisation in order to achieve the targets. The key challenge for an optimisation algorithm is to deliver a good solution in a very short time. Furthermore, the found configuration must be stable enough to sustain sufficiently long to avoid continuous migration actions of VMs across the data centre.

There are a variety of optimization algorithms that are used in VM placement scenario, such as Bin packing problem, Convex optimization problem, Minimum K cut/balanced Minimum K cut, Knapsack problem and Metaheuristic-based approaches such as Ant Colony Optimization (ACO), Simulated Annealing (SA) etc. However, a library of a set of cloud specific optimization algorithms is missing in Python. For testing purposes, one needs to manually implement the algorithms which is often time consuming.

Scope of the Thesis

The thesis requires at first an in-depth study about the optimization algorithms/techniques which are currently being used in placing VMs in a cloud infrastructure. The following objective functions should be considered while selecting the suitable algorithms

1. Server minimization
2. Energy and operational cost reduction
3. Resource usage maximization

The ultimate goal is to find a set of best suited algorithms and integrate them into a framework which can be used to validate any VM placement algorithm.

Requirements and Comments

1. Good knowledge in Python is a must.
2. Knowledge about cloud network and resource management

If you are interested, please contact Stefan Wesner and Mitalee Sarker either by mail or directly in the office.
Mail: stefan.wesner@uni-ulm.de, office: Uni West, 43.2.210
mitalee.sarker@uni-ulm.de office: Uni West, 43.2.220

Faculty of Engineering and Computer Science

Institute of Information Resource Management