CCO - Cloud Cost Optimization

Adi Yehoshua
Red Hat
Ra'anana, Israel
ayehoshu@redhat.com

Ilya Kolchinsky
Red Hat
Ra'anana, Israel
ikolchin@redhat.com

Assaf Schuster
Technion
Haifa, Israel
assaf@technion.ac.il

ABSTRACT

Cloud computing can be complex, but optimal management of it doesn't have to be. In this paper, we present the design and implementation of a scalable multi-Cloud Cost Optimizer (CCO) that calculates the optimal deployment scheme for a given workload on public or hybrid clouds. The goal of CCO is to reduce monetary costs while taking into account the specifications of the workload, including resource requirements and constraints. By using a combination of meta-heuristics, CCO addresses the combinatorial complexity of the problem and currently supports AWS and Azure. The CCO tool [1], can be accessed through a web UI or API and supports on-demand and spot instances.

CCS CONCEPTS

• **Computer systems organization** → **Cloud computing**; • **Theory of computation** → Optimization with randomized search heuristics; • **Information systems** → Cloud computing performance evaluation.

KEYWORDS

Cost optimizer, Cloud computing, AWS, Azure, Hybrid cloud

1 INTRODUCTION

Cloud computing has transformed the way organizations consume computing resources, offering greater flexibility, scalability, and accessibility. However, the management of cloud costs has emerged as a significant challenge. Public clouds have become popular due to their scalability and cost-effectiveness for complex workloads. Nevertheless, selecting, managing, and monitoring cloud services requires careful consideration, as they can be costly, especially for large and complex workloads. Furthermore, the complexity and the magnitude of public cloud offerings presents a formidable challenge with substantial combinatorial complexity that makes it increasingly difficult for organizations to make informed decisions about cloud deployment. With thousands of possible virtual machines across various instance types, regions, and operating systems, finding the most cost-effective solution can quickly become impractical and even impossible. Therefore, a "brute force" approach of examining all possible alternatives and selecting the cheapest option is highly impractical, especially for applications requiring a large number of VMs.

To tackle this challenge, we propose a novel tool, CCO (Cloud Cost Optimizer). Unlike cloud brokers [3], which manages the use, performance, and delivery of cloud services, CCO is designed to minimize the monetary costs associated with deploying complex workloads on public clouds. CCO is the first tool to search through the *entire range of solutions* available from each cloud provider to find the heuristically optimal solution to launch. This makes CCO an innovative solution to the problem of reducing cloud costs. At present, CCO supports AWS and Azure and can easily integrate with other public clouds through its plugin interface. Furthermore, CCO fully supports on-demand/pay-as-you-go and spot instances, which can help customers reduce instance costs by up to 90%.

2 ALGORITHM

CCO takes in a specification of the desired workload and calculates the minimum cost scheme for deploying the application on a public cloud. To overcome the combinatorial complexity of the problem, the tool uses a combination of meta-heuristics, such as *tabu search* [2] and *simulated annealing* [4]. This results is a solution that minimizes the expected monetary cost of deployment. The workload specifications that are provided to CCO include the resource requirements for each component of the workload, as well as the connections between the components and any additional constraints that must be taken into account.

3 RESULTS

The study's results indicate that when dealing with a small number of components (specifically, nine components as displayed in Figure 1), the heuristic algorithm is unable to locate the optimal solution after five seconds, in contrast to the brute force method which takes less than a second to find the optimal solution. However, the difference between the two approaches is not significant, with less than 0.01 USD. In contrast, as the number of components increases (as demonstrated in Figure 2, where twenty components are analyzed), the heuristic algorithm outperforms the brute force method by producing superior outcomes in less than two seconds, ultimately achieving results that are twice as good within five seconds. These findings imply that, while the brute force method may be more appropriate for certain scenarios, the heuristic algorithm may be a more practical approach when working with a larger quantity of components.

SYSTOR'23, June 2023, Haifa, Israel

ACM ISBN 978-1-4503-9962-3/23/06…$15.00
https://doi.org/10.1145/3579370.3594746

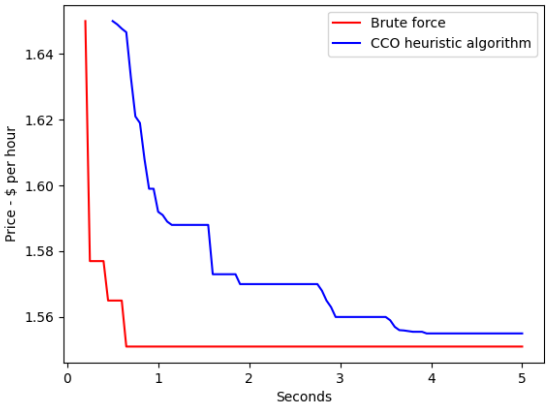

Figure 1: Algorithm comparison - 9 components.

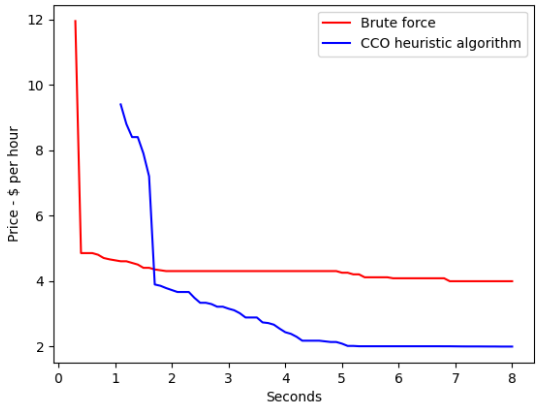

Figure 2: Algorithm comparison - 20 components.

4 FUTURE WORK

The functional version of CCO is already available for use [1]. Yet, there are many potential areas for further expansion and improvement. Some examples include replacing the current metaheuristic-based optimization algorithm with a reinforcement learning approach, enabling advanced planning through forecasting of instance prices based on market conditions, incorporating future interruption rates of spot instances into CCO's calculations, and adding support for reserved instances.

REFERENCES

[1] https://tinyurl.com/mw57knrj.
[2] Michael Gendreau. 2003. An introduction to tabu search. In *Handbook of metaheuristics*. Springer US, 37–54.
[3] Xingjia Li, Li Pan, and Shijun Liu. 2022. A survey of resource provisioning problem in cloud brokers, In Journal of Network and Computer Applications. *Journal of Network and Computer Applications* 203, 103384.
[4] Alexander Nikolaev and Sheldon Jacobson. 2010. Simulated annealing. In *Handbook of metaheuristics*. Springer US, 1–39.