UPPSALA
UNIVERSITET

Benchmarking of option price solvers as a service in OpenStack

Applied Cloud computing project(1TD265)
Project Report by Team 4

Melker Forssell, Markus Skogsmo, Isak Borg, Shuzhi Dong

October 23, 2019

Abstract

In this report, we present the integration of an existent project BENCHOP which is a collaborative project initiated by Computational Finance research group from Uppsala university to a Cloud Computing environment by developing a cloud service to support users running the benchmark. By using contextualization feature with Docker, this integration enable launching the application as a public service on cloud.

1 Introduction

1.1 Project description

This project is to develop a cloud service to allow more users to run and utilize the benchmark. Due to licensing issues GNU Octave [2], which is an open source project and implementation of the Matlab programming language, has been chosen for this project.

The goal of this project is to implement this cloud based system as a service so that users can, through a web-interface, start the different benchmarks of option price solvers provided by the BENCHOP project. BENCHOP provides a common suite of benchmark problems for option pricing [1], which is done to provide the finance community with tools that can be used both for comparisons between existing methods and for evaluation of new methods.

Black-scholes Model is a pricing model used to determine the fair price or theoretical value for a call or a put option based on six variables: volatility, type of option, underlying stock price,time, strike price, and risk-free rate. The quantum of speculation a more in case of stock market derivatives, and hence proper pricing of options eliminates the opportunity for any arbitrage. This model is used to determine the price of a European call option, which simply represents the option can only be exercised on the expiration date. In mathematical formulation [1],we let S represent the actual(stochastic) asset price realization, s is the asset price in the PDE formulation, t is the time,r is the risk free interest rate, σ is the volatility,W is a Wiener process, u is the option price.

SDE-setting:

$$dS = rSdt + \sigma SdW$$

PDE-setting:

$$\frac{\partial u}{\partial t} + \frac{1}{2}\sigma^2 s^2 \frac{\partial^2 u}{\partial s^2} + rs\frac{\partial u}{\partial s} - ru = 0$$

On the whole, with different numerical method, different option types can be solved.In our application, benchmarks performed by European call option(Problem 1a),American call option(Problem 1b)and Up-and-Out option(Problem 1c). In our app.py file, we can choose which problem to fix by changing the .....

Three numerical methods, the COS method, finited difference(FD) method and RBF-FD method have been test in our application.

1.2 Problem description

We will be using Docker and dockerswarm to create as many virtual machines that we require to complete the time intensive benchmark problems in a reasonable time. To create the task queues and the different workers required, we are using a celery/redis setup, and we use flask to interact with the celery worker.

We take the docker container(s), then dockercompose to connect them and create an application of containers, then make online an, cloud based, services out of them using dockerswarm (orchestration tool used to create and manage a swarm).

2 Model

Our main architecture includes 4 parts: Web API,a Redis server, flask framework,three celery workers.

2.1 Server

2.2 Docker setup

2.3 contextualization

3 Results

3.1 Main program

3.2 Scalability

4 Conclusion and Discussion

References

[1] L. Höök, E. Larsson *et al.*, "BENCHOP—The BENCHmarking project in Option Pricing," *International Journal of Computer Mathematics*, 2015. [Online]. Available: http://uu.diva-portal.org/smash/get/diva2:848689/FULLTEXT01.pdf?fbclid=IwAR268VCJxioxza8ARPOO9YAnm_OBjDHd5z9dnKdT1jJ3zEap1TE6oPc0ezQ

[2] GNU Octave. Octave. Accessed 2019-10-20. [Online]. Available: https://www.gnu.org/software/octave/