Automated Benchmarking of Container Applications

Paulius Dilkas

9th July 2019

1 Introduction

2 Architecture

2.1 Software Architecture

2.2 Software Configuration

Added to the Flink configuration file (`flink-conf.yaml`):

```
metrics.latency.interval: 1000
metrics.reporters: prom
metrics.reporter.prom.class: org.apache.flink.metrics.prometheus.PrometheusReporter
metrics.reporter.prom.port: 9250
```

Prometheus is configured as follows (`prometheus.yml`):

```
global:
  scrape_interval:     1s
  evaluation_interval: 1s

scrape_configs:
  - job_name: 'benchmarker'
    static_configs:
      - targets: ['jobmanager:9250', 'taskmanager:9250']
```

Modified the configuration file of the Prometheus add-on for MiniShift[1] to disable OAuth-based authentication by replacing `-skip-auth-regex=^/metrics` with `-skip-auth-regex=^/`.

3 Docker and OpenShift

Two simple Dockerfiles were created.
Control pod entrypoint:

```
#!/bin/sh

flink run -m jobmanager:8081 benchmarker-0.1.jar &
java -cp benchmarker-0.1.jar ControlServer
```
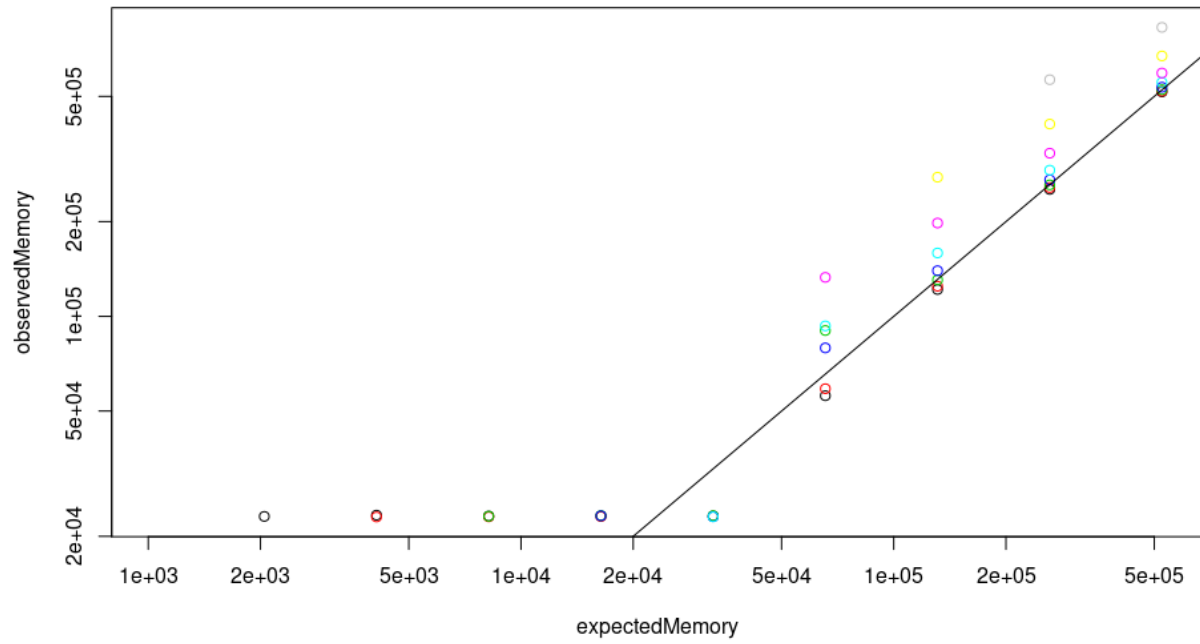
---

[1] `https://github.com/minishift/minishift-addons/tree/master/add-ons/prometheus`

Figure 1:

4 Local Performance Tuning
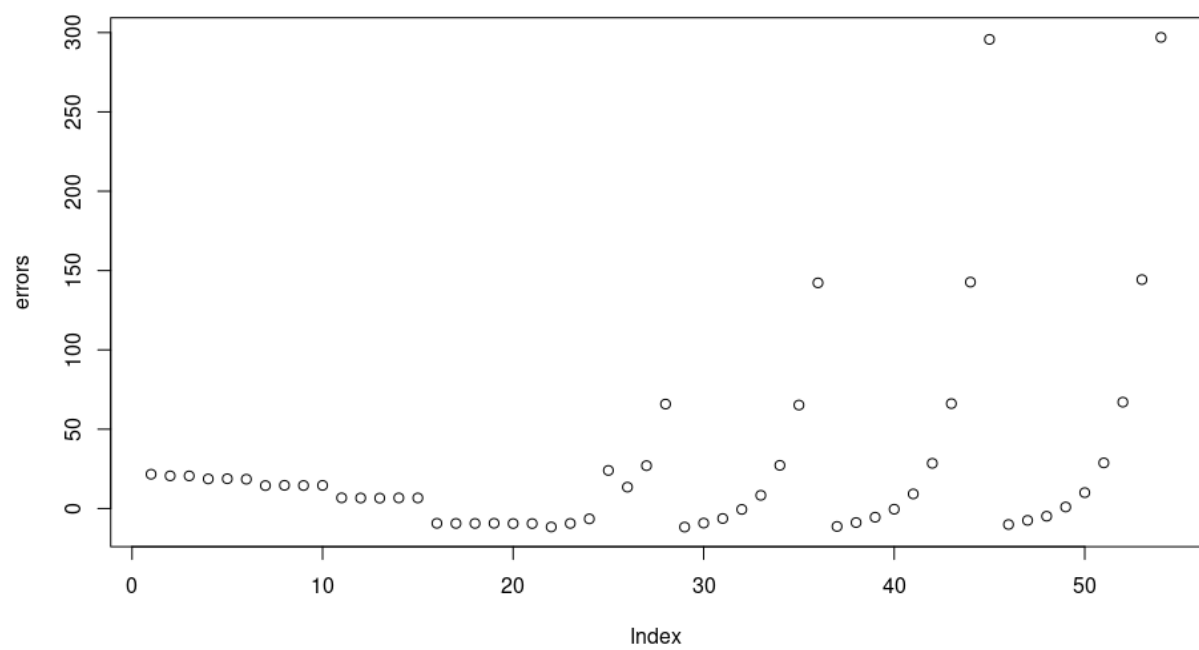
4.1 Experiments

4.2 Adjusted Performance

5 Experimental Results

6 Conclusion

Figure 2:

Figure 3:

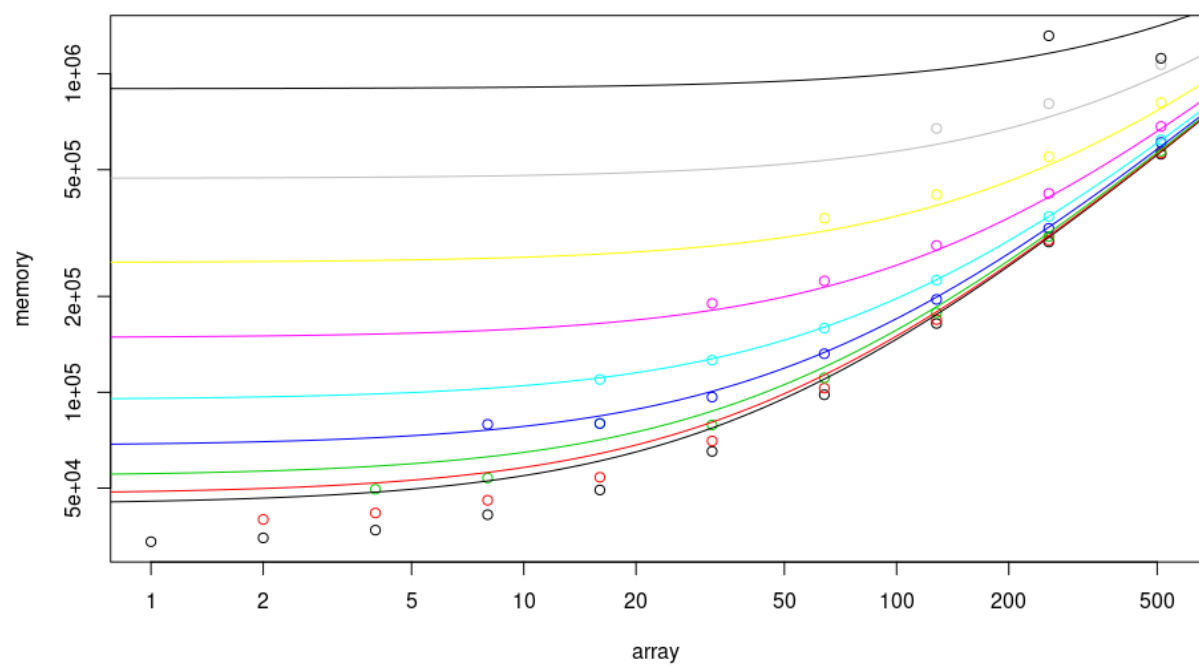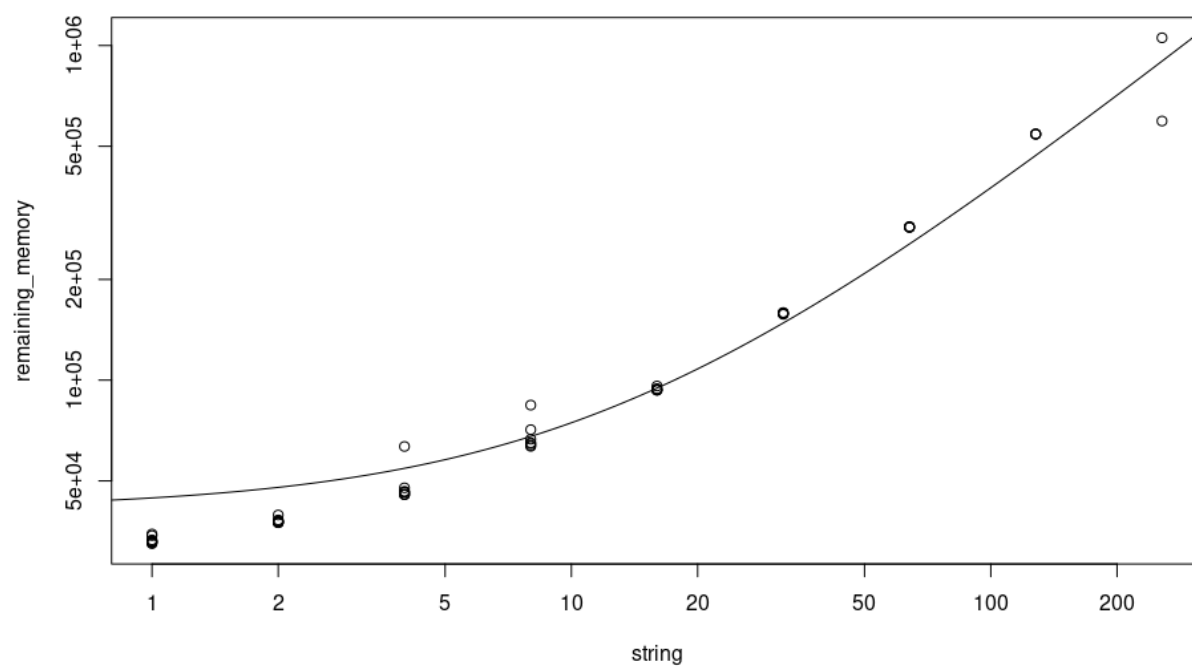Figure 4: