

华中科技大学

数据中心技术课程实验报告

院 系 计算机科学与技术学院

班 级 2107

学 号 M202173752

姓 名 刘一航

2021 年 12 月 31 日

一、 实验环境搭建

执行 run-minio.cmd 命令搭建了 MinIO 服务器，取得服务器地址。用设定的用户名和密码通过 MinIO 给出的服务器地址登录。在新创建的 MinIO 服务器中新建一个名为“loadgen”的 Bucket，完成系统搭建和预备操作。

```
PS C:\Users\tom\Desktop\minio\obs-tutorial> .\run-minio.cmd
API: http://211.69.197.153:9000 http://192.168.192.1:9000 http://192.168.77.1:9000 http://127.0.0.1:9000
RootUser: hust
RootPass: hust_obs

Console: http://211.69.197.153:9090 http://192.168.192.1:9090 http://192.168.77.1:9090 http://127.0.0.1:9090
RootUser: hust
RootPass: hust_obs

Command-line: https://docs.min.io/docs/minio-client-quickstart-guide
$ mc.exe alias set myminio http://211.69.197.153:9000 hust hust_obs

Documentation: https://docs.min.io

+-----+
| You are running an older version of MinIO released 2 weeks ago |
| Update:                                                         |
| Run `mc admin update`                                           |
+-----+
```

```
PS C:\Users\tom\Desktop\minio\obs-tutorial> ./mc mb root/loadgen
Bucket created successfully 'root/loadgen'.
PS C:\Users\tom\Desktop\minio\obs-tutorial> ./mc mv test.txt root/loadgen
test.txt: 12 B / 12 B [-----] 3.27 KiB/s 0s
PS C:\Users\tom\Desktop\minio\obs-tutorial> ./mc cat root/loadgen/test.txt
hello world!
PS C:\Users\tom\Desktop\minio\obs-tutorial> ./mc rb -force root/loadgen
Removed 'root/loadgen' successfully.
```

二、 性能测试

执行 run-s3bench.cmd 命令开始 s3bench 基准测试。

```
Test parameters
endpoint(s): [http://211.69.197.153:9000]
bucket:      loadgen
objectNamePrefix: loadgen
objectSize:  0.0010 MB
numClients:  8
numSamples:  256
verbose:     %!d(bool=false)
```

```
Results Summary for Write Operation(s)
Total Transferred: 0.249 MB
Total Throughput:  0.16 MB/s
Total Duration:    1.598 s
Number of Errors:  1

-----
Write times Max:      0.180 s
Write times 99th %ile: 0.173 s
Write times 90th %ile: 0.077 s
Write times 75th %ile: 0.062 s
Write times 50th %ile: 0.047 s
Write times 25th %ile: 0.030 s
Write times Min:      0.007 s
```

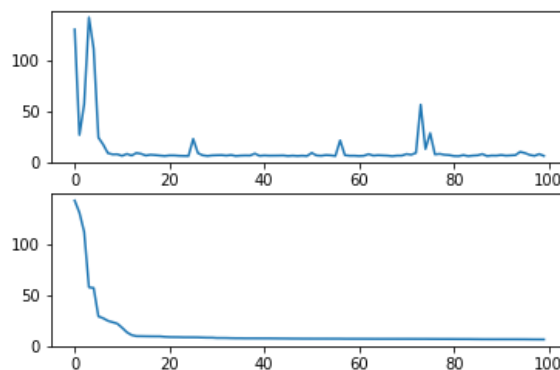
```
Results Summary for Read Operation(s)
Total Transferred: 0.249 MB
Total Throughput: 7.15 MB/s
Total Duration: 0.035 s
Number of Errors: 1
-----
Read times Max: 0.003 s
Read times 99th %ile: 0.003 s
Read times 90th %ile: 0.002 s
Read times 75th %ile: 0.001 s
Read times 50th %ile: 0.001 s
Read times 25th %ile: 0.001 s
Read times Min: 0.000 s
```

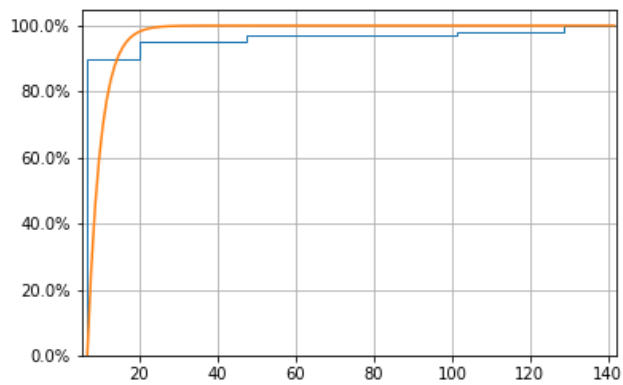
分析图中数据可知, 最大时长往往比最小时长高出许多倍, 同时也大大超出 90%基准线的标准, 这些占比很小的数据传输很多时候会造成额外的时间成本。

三、尾延迟挑战

1. 普通请求

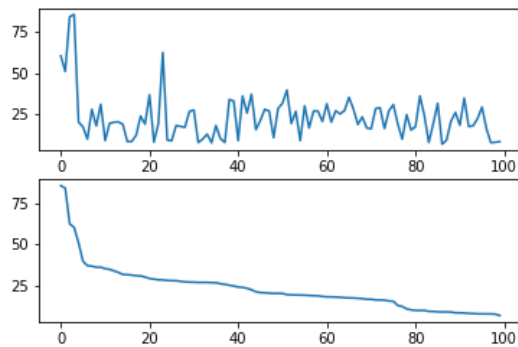
从以下实验数据可以观测到, 最小延迟为 6.19 ms, 平均延迟为 15.85 ms, 50%的请求在 8.81 ms 内完成, 95% 的请求在 34.66 ms 内完成, 99% 的请求在 101.31 ms 内完成, 最大延迟为 164.85 ms



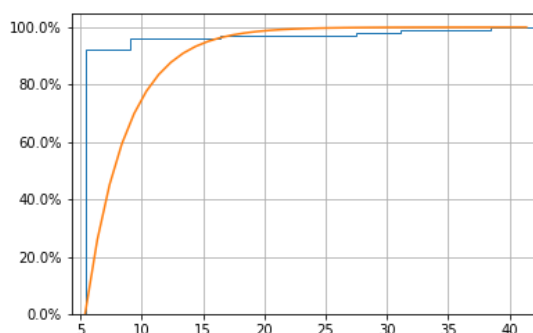


2.对冲请求

创建对冲请求结果如下，对比普通请求的延迟情况可知，普通请求在头部延迟上有优势，而对冲请求在尾部延迟上有优势



。



3.关联请求

优化后重新测试结果如下，分析数据可发现，延迟分布相比之前更加均匀，数据的方差和极值均显著降低，可见尾延迟优化效果明显。

