Distributed Shell

Jinan Hu, Zachary Halzel

April 5, 2018

1 Design

We set up an Amazon cloud server instance and ran experiments to measure

1) Network throughput and latency
2) CPU and I/O performance

Then, we used a simple algebraic model to predict how much data would need to be transferred for cloud computing to be more efficient than local processing.

1.1 CPU Performance

To perform the cpu test, we ran the following command on the local VM and the Amazon VM:
> sysbench –test=cpu –cpu-max-prime=20000 run

1.2 File I/O Performance

We ran the following commands on the local VM as well as the Amazon VM:
> sysbench –test=fileio –file-total-size=2G prepare
> sysbench –test=fileio –file-total-size=2G –file-test-mode=rndrw –max-time=30 –max-requests=0 run

1.3 Network Performance

Latency of connection (milliseconds)

To test the amount of time required to setup a connection to the server, authenticate, and tear down, we added a test option in both the client and server side code. This option creates and authenticates a connection but does not run any commands, in order to isolate the processes we want to time. The test option appends the duration in milliseconds of the operation to a text file, and we have a bash script that runs this test as many times as we want (we chose 100) so that we could generate a sample and derive the mean and standard deviation of the data.

Maximum Throughput

To measure the maximum throughput, we used scp to send files of various sizes from the EC2 server to our machine. The scp command conveniently outputs the throughput in MB/second.

2 Results

2.1 Network Performance

Connection Latency

As seen in Figure 2, after 100 trials the connection time to the Amazon EC2 server tended to hover between 80 and 100 milliseconds, with a few outliers.

Network Throughput

As seen in table 1, the network throughput ranged from approximately 10 to 14 MB/Second over a series of trials with varying file sizes. The mean throughput is 12.25 with a standard deviation of 1.23

2.2 CPU Performance

Local VM

events per second: 320.56
total time: 10.0021s
total number of events: 3207

Amazon VM

events per second: 296.67
total time: 10.0011s
total number of events: 2967

Local VM Latency (ms)

min: 2.80
avg: 3.12
max: 6.63
95th percentile: 4.74
sum: 9997.37

Amazon VM Latency (ms)

min: 3.07
avg: 3.37
max: 3.70
95th percentile: 3.43
sum: 9995.28

Local VM Threads fairness

events (avg/stddev): 3207.0000/0.00
execution time (avg/stddev): 9.9974/0.00

Amazon VM Threads fairness

events (avg/stddev): 2967.0000/0.00
execution time (avg/stddev): 9.9953/0.00

2.3 File I/O Comparison

Local VM File operations

reads/s: IOPS=1228.69
writes/s: IOPS=819.13
fsyncs/s: IOPS=2619.50

Amazon VM File operations

read: IOPS=1277.98
write: IOPS=851.99
fsync: IOPS=2722.66

Local VM Throughput

read, MiB/s: 19.20
written, MiB/s: 12.80

Amazon VM Throughput

read, MiB/s: 19.97
written, MiB/s: 13.31

Local VM Latency (ms)

min: 0.00
avg: 0.21
max: 38.46
95th percentile: 0.62
sum: 29814.54

Amazon VM Latency (ms)

min: 0.00
avg: 0.20
max: 5.43
95th percentile: 0.67
sum: 29768.72

2.4 Model

$$cpu + local_file_io = n * Network + remotecpu + remotefileio$$

$$Network = 6.67 * 10^-8 s/byte$$

$$localcpu = 10.0021s$$

$$localio = 16GB/16Mb/s = 1024s$$

$$remotecpu = 10.0011s$$

$$remoteip = 16GB/16.64Mb/s = 984.6s$$

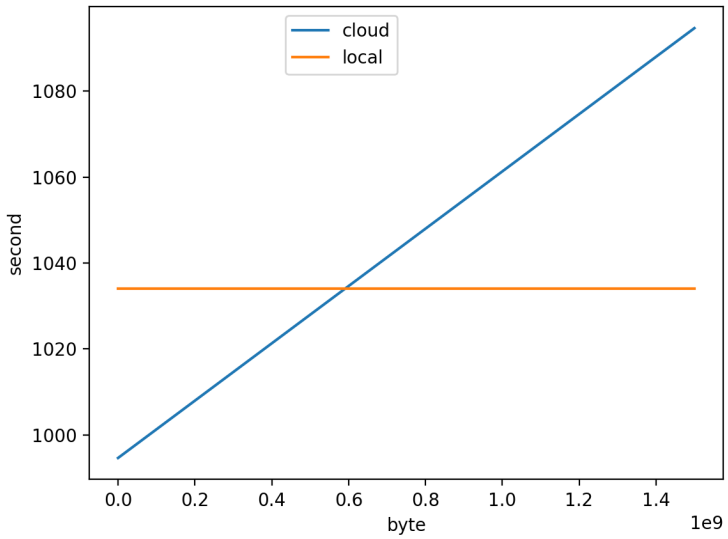
$$n = 5.9 * 10^8 byte$$

3 Analysis

Local vs. remote

As seen in the graph (figure 1), local computing is more efficient up to the point where the lines intersect. At this junction, cloud computing overtakes local computing in terms of efficiency. This simplified model demonstrates the effectiveness of the distributed computing paradigm as the magnitude of data in society increases exponentially.

Figure 1: Local vs. Cloud computing



File Size	MB/Sec	bits/Sec
100 MB	10	83886080
200 MB	14.3	119957094.4
300 MB	11.1	93113548.8
400 MB	11.1	93113548.8
500 MB	12.5	104857600
600 MB	12.5	104857600
700 MB	13.2	110729625.6
800 MB	13.8	115762790.4
900 MB	13.0	109051904
1000 MB	13.2	110729625.6

Table 1: File Size vs. Throughput

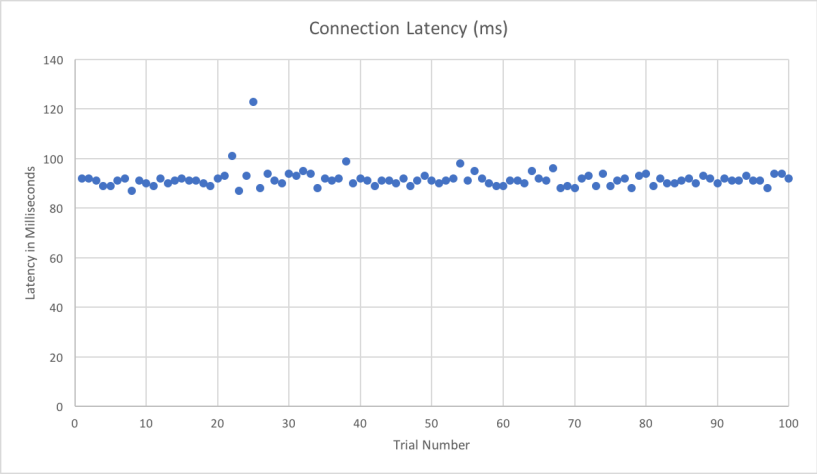


Figure 2: Latency of connection time in milliseconds