CS 553
CLOUD COMPUTING
PROGRAMMING ASSIGNMENT-3
CloudKon clone with Amazon EC2, S3, SQS, and DynamoDB

Performance Evaluation

SUBMITTED BY :
SACHIN KRISHNA MURTHY
CWID : A20354077

Performance Evaluation

Local Back-end Workers

Formulas Used :

Ideal Time = (Sleep Time * Number of tasks) / (Number of workers)

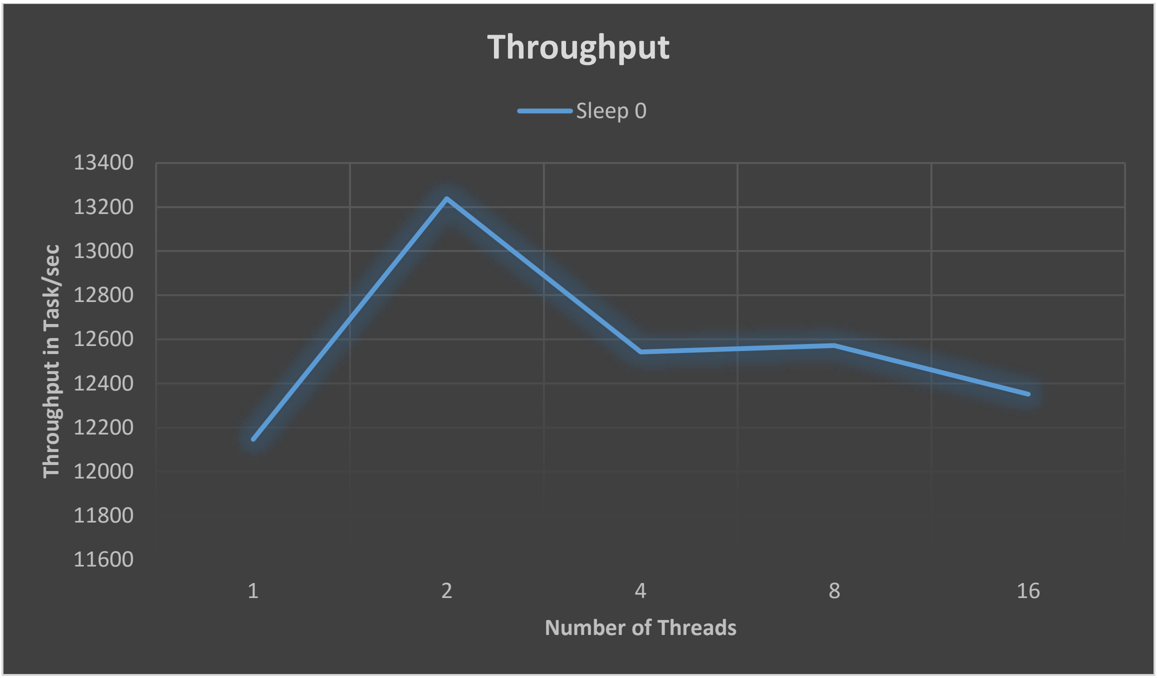
Throughput = (Number of Tasks) / (Number of Workers)

Efficiency = (Ideal Time) / (Actual Time Taken)

- **SLEEP 0**

SLEEP 0	LOCAL WORKERS	
Number of Threads	Time Taken (in sec)	Throughput (Task/sec)
1	8.233	12146.24074
2	7.554	13238.01959
4	7.973	12542.33036
8	7.954	12572.29067
16	8.096	12351.77866

- **Throughput :**



From the above graph, we can notice that the throughput value increases from 1 thread to 2 thread and goes not decreasing further for 8 and 16 threads.

- **SLEEP 10ms**

SLEEP 10ms						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput (Task/sec)	Efficiency in percentage
1	1000	10.37	10	0.964320154	96.43201543	96.43201543
2	2000	10.48	10	0.954198473	190.8396947	95.41984733
4	4000	10.562	10	0.946790381	378.7161522	94.67903806
8	8000	10.729	10	0.932053313	745.6426508	93.20533134
16	16000	10.97	10	0.911577028	1458.523245	91.15770283

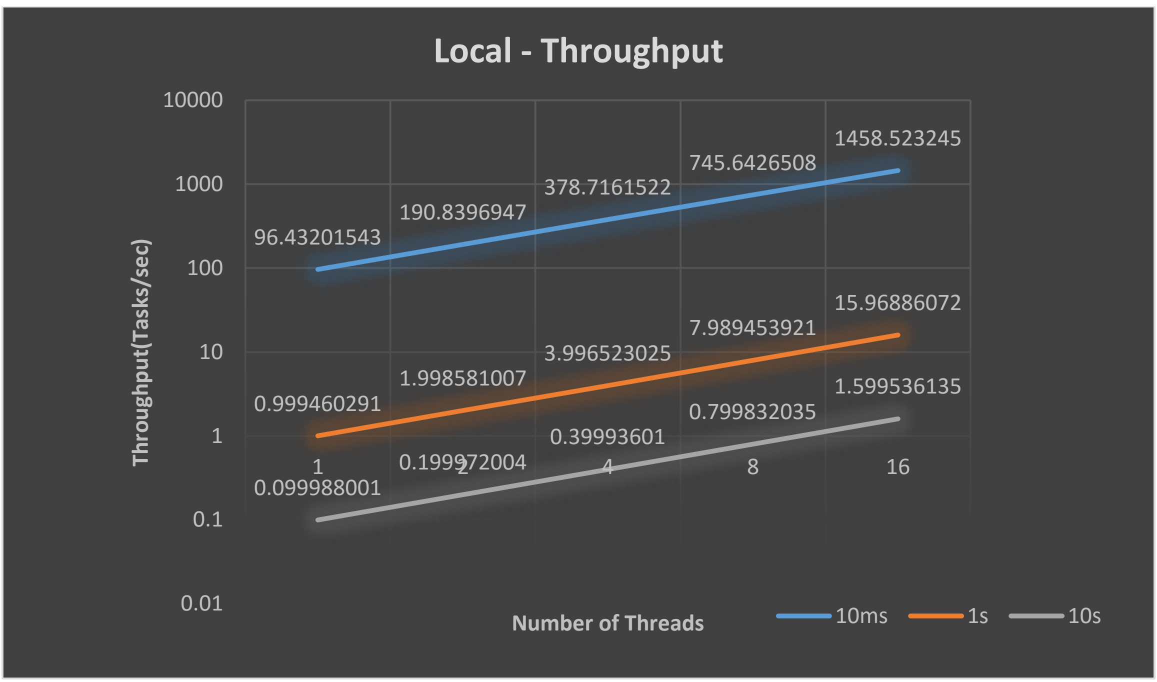
- **SLEEP 1sec (1000ms)**

SLEEP 1 sec (1000 ms)						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput (Task/sec)	Efficiency in percentage
1	100	100.054	100	0.999460291	0.999460291	99.94602914
2	200	100.071	100	0.999290504	1.998581007	99.92905037
4	400	100.087	100	0.999130756	3.996523025	99.91307562
8	800	100.132	100	0.99868174	7.989453921	99.86817401
16	1600	100.195	100	0.998053795	15.96886072	99.80537951

- **SLEEP 10sec (10000ms)**

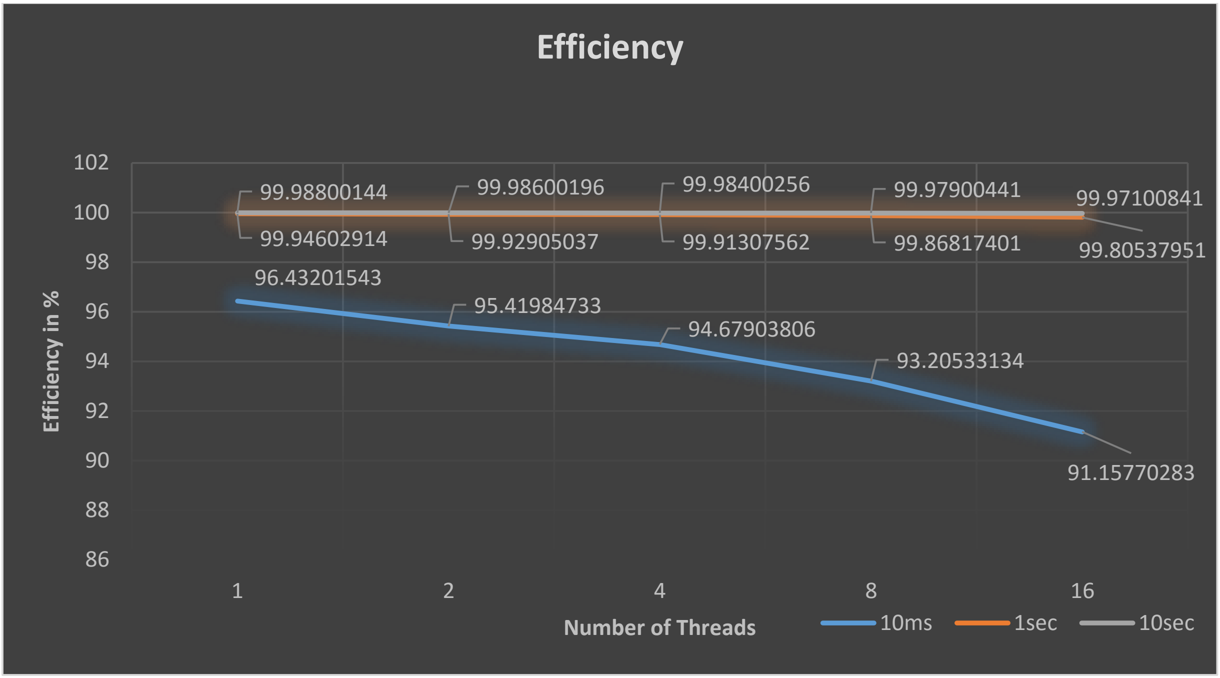
SLEEP 10 sec (10000 ms)						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput (Task/sec)	Efficiency in percentage
1	10	100.012	100	0.999880014	0.099988001	99.98800144
2	20	100.014	100	0.99986002	0.199972004	99.98600196
4	40	100.016	100	0.999840026	0.39993601	99.98400256
8	80	100.021	100	0.999790044	0.799832035	99.97900441
16	160	100.029	100	0.999710084	1.599536135	99.97100841

- **Throughput :**



The above graph gives us the details of the throughput values of the local back-end workers.From the above graph, we can observe that there is a linear increase in the values of throughput for threads 1,2,4,8,16 for 10ms, 1sec and 10 sec.

- **Efficiency**



The above graph shows the efficiency of the local back-end workers. From the above efficiency graph, we can notice that the efficiency does not vary by a large margin for different number of threads i,e for 1,2,4,8 & 16 threads and for 1sec and 10 sec and decreases for 10 ms from 1 thread to 16 threads.

So we can say that as the number of workers increases, task scheduling becomes difficult which makes the workers to remain idle and hence efficiency decreases.

Remote Back-End Workers

- **SLEEP 10ms**

REMOTE WORKERS						
SLEEP 10ms						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput (Task/sec)	Efficiency in percentage
1	1000	155.047	5	0.032248286	6.449657201	3.2248286
2	2000	153.675	5	0.032536197	13.01447861	3.253619652
4	4000	479.593	5	0.010425507	8.340405302	1.042550663
8	8000	1197.396	5	0.004175728	6.681164794	0.4175728
16	16000	2817.095	5	0.001774878	5.679609669	0.177487802

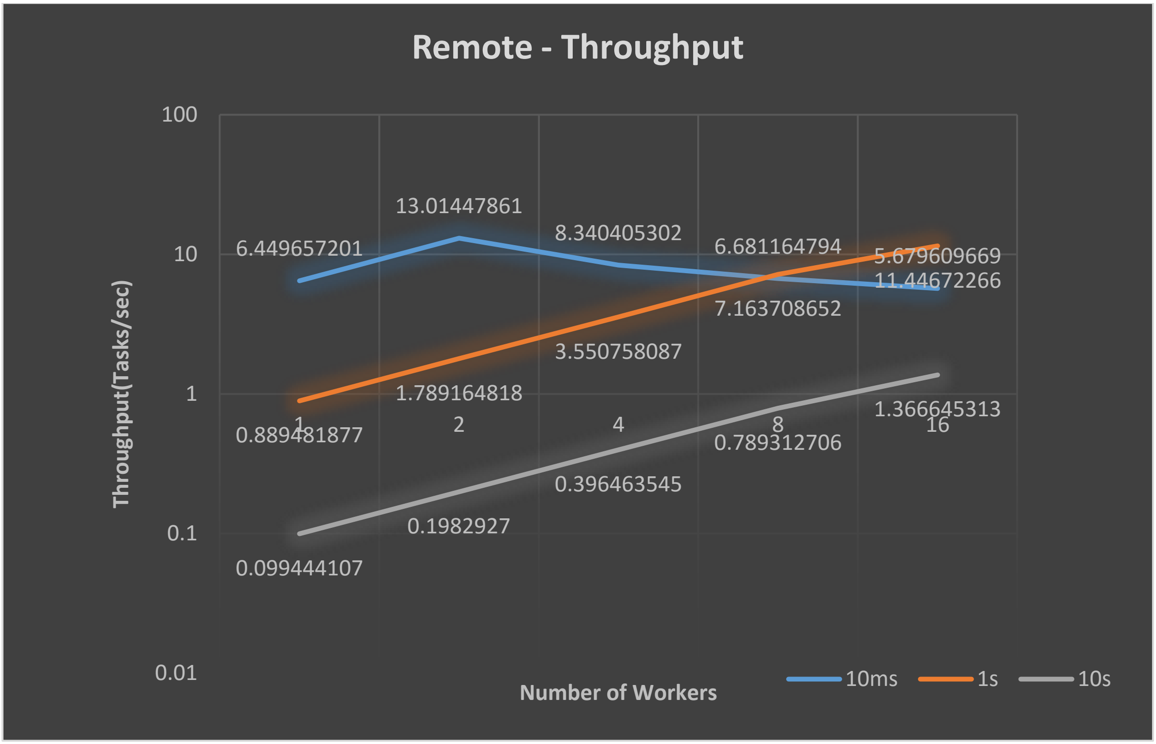
- **SLEEP 1sec (1000ms)**

SLEEP 1 sec (1000 ms)						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput(Task/sec)	Effciency in percentage
1	100	112.425	50	0.444740938	0.889481877	44.47409384
2	200	111.784	50	0.447291204	1.789164818	44.72912045
4	400	112.652	50	0.443844761	3.550758087	44.38447609
8	800	111.674	50	0.447731791	7.163708652	44.77317907
16	1600	139.778	50	0.357710083	11.44672266	35.77100831

- **SLEEP 10sec (10000ms)**

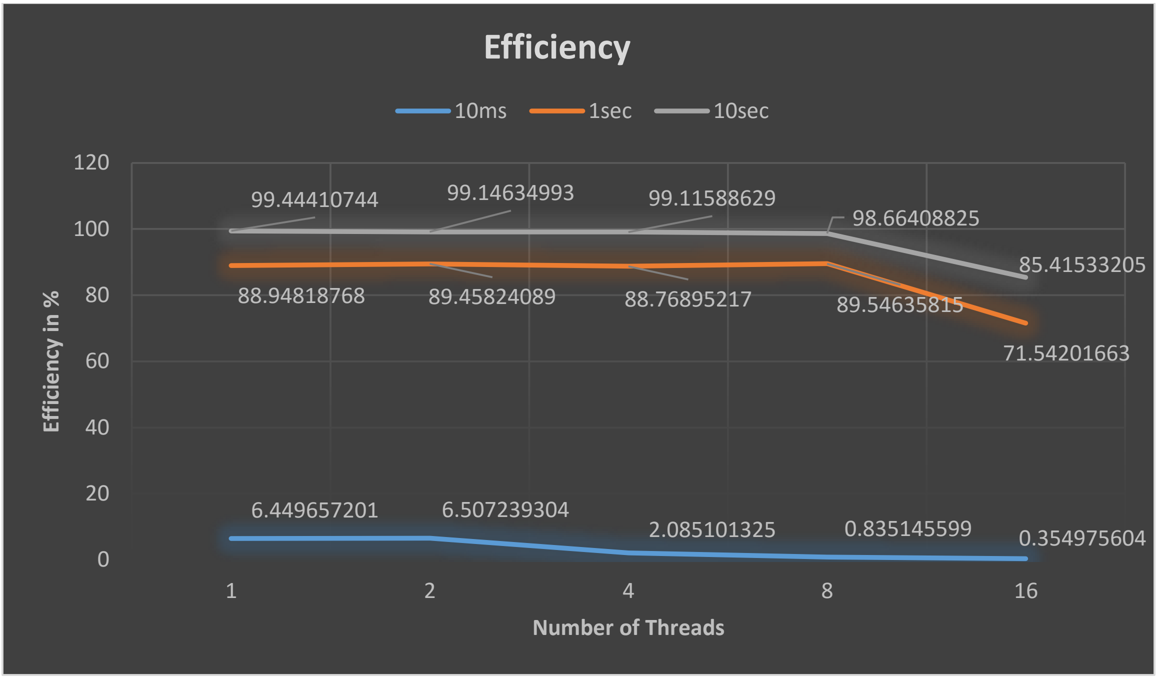
SLEEP 10 sec (10000 ms)						
Number of Threads	Number of Tasks	Time Taken (in sec)	Ideal Time	Efficiency	Throughput (Task/sec)	Effciency in percentage
1	10	100.559	50	0.497220537	0.099444107	49.72205372
2	20	100.861	50	0.49573175	0.1982927	49.57317496
4	40	100.892	50	0.495579431	0.396463545	49.55794315
8	80	101.354	50	0.493320441	0.789312706	49.33204412
16	160	117.075	50	0.42707666	1.366645313	42.70766603

- **Throughput**



Here, we can say that the throughput values for remote back-end workers for 1sec and 10 sec increases linearly for different number of threads whereas for 10 ms we can see that the throughput value increases for 1 thread and 2 threads and goes on decreasing for 4,8 and 16 threads.

- **Efficinecy :**



The above graph gives us the information regarding the efficiency of the remote workers.From the above graph, we can notice that the efficiency value does not vary much for 1 sec and 10 sec from 1thread to 8 threads and decreases for 16 threads. For 10 ms, the efficiency value does not vary much for 1 and 2 threads but decreases for 4,8 and 16 threads.

So we can say that as the number of workers increases, task scheduling becomes difficult which makes the workers to remain idle and hence efficiency decreases.