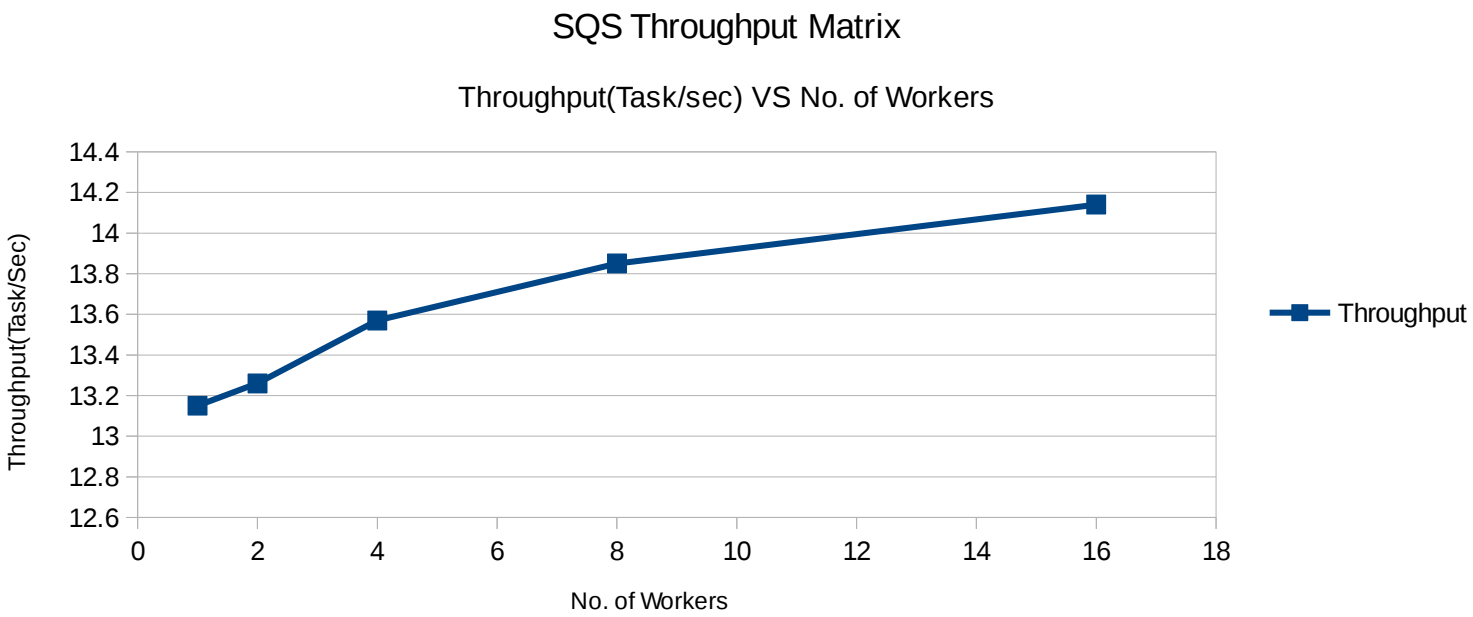
Performance Evaluation

- **Remote Client-Worker :**
  - **Throughput:**



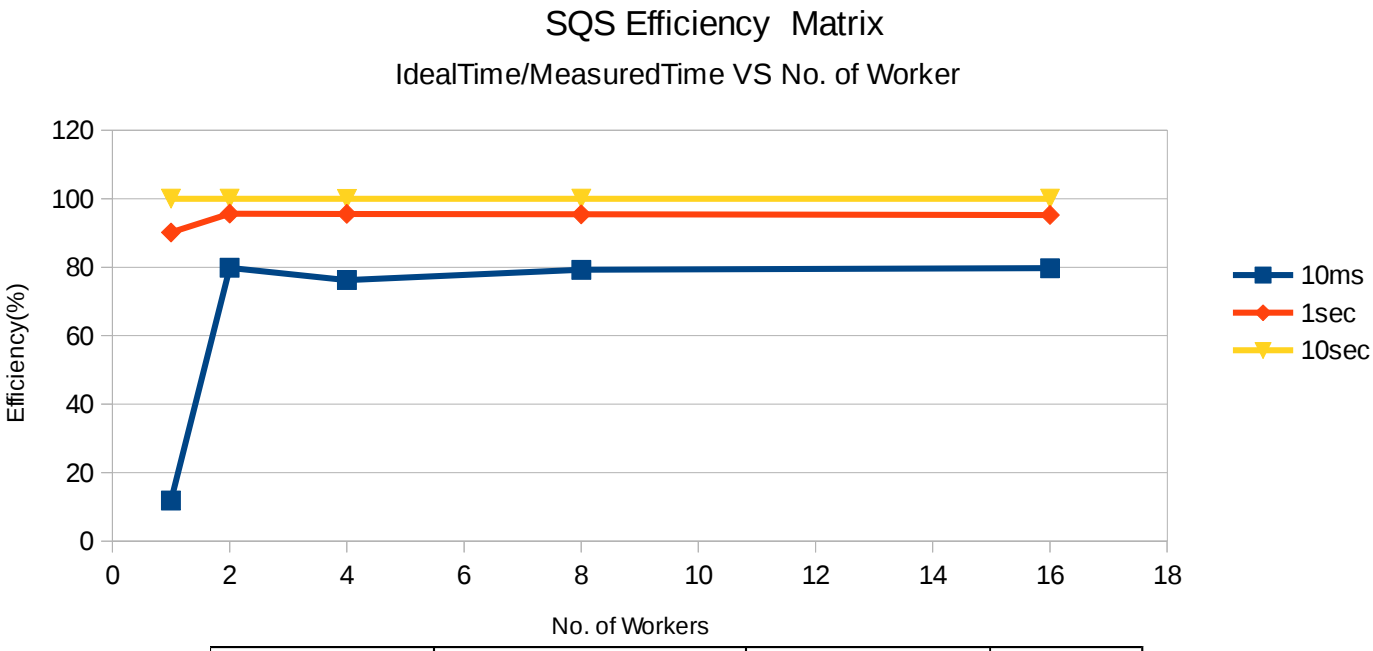
No. Of Worker	Throughput	Measured Time	Tasks
1	13.15	760.55	10000
2	13.26	754.143	10000
4	13.57	736.895	10000
8	13.85	721.843	10000
16	14.14	707.152	10000

- The above graph shows the throughput calculation of Amazon SQS through 100k Sleeep(0) operations. The graph is drawn for the variance of No. of Workers(Amazon EC2 t2.micro instances) with the variance value of 1, 2, 4, 8 and 16 with their throughput value.

Throughput= Tasks / Measured Time

- From the graph it is shown that throughput increases as the number of the workers doubled. Reason for this is time spent by the workers in synchronization. However, throughput is not doubling with it but it is increasing linearly.

- **Efficiency:**

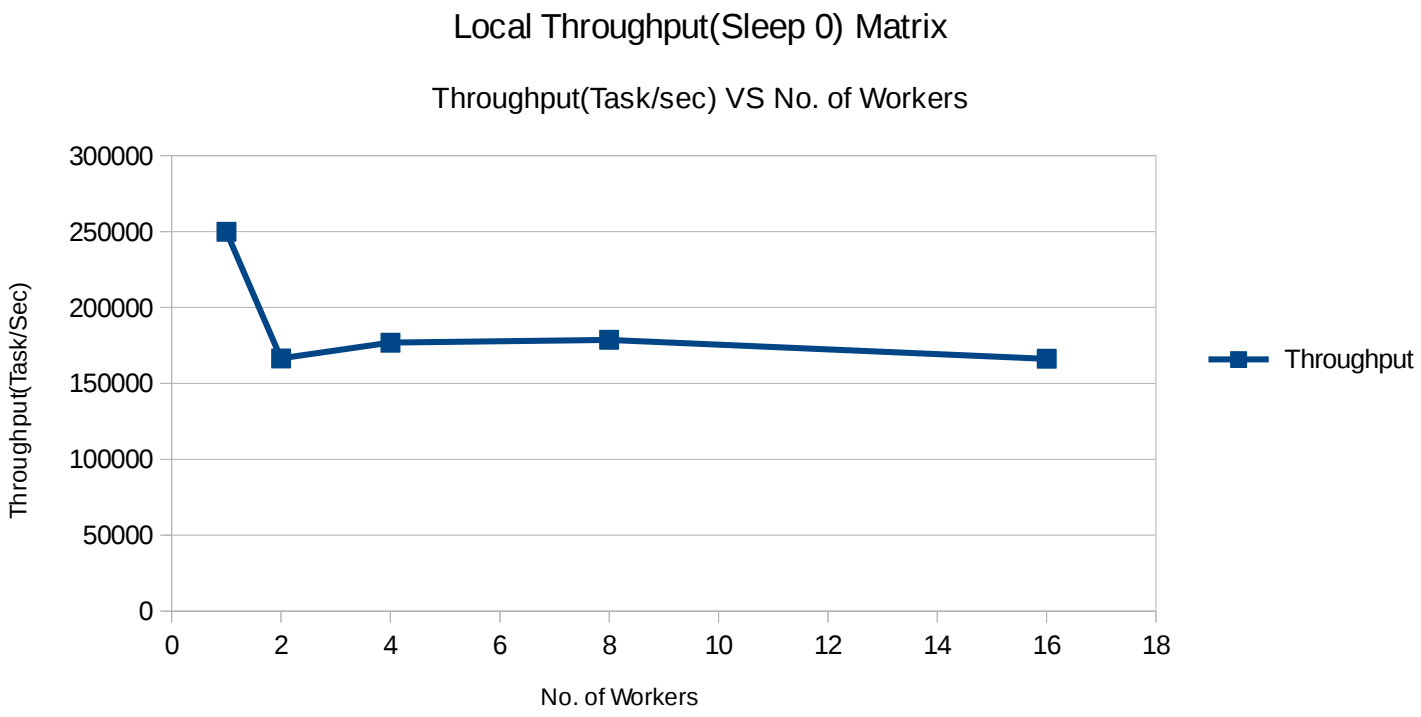


No. Of Worker	10ms	1sec	10sec
1	11.88	90.14	99.99
2	79.83	95.65	99.98
4	76.25	95.55	99.97
8	79.23	95.44	99.98
16	79.71	95.26	99.98

- The above graph shows the efficiency calculation of Amazon SQS workers. The graph is drawn for the variance of No. of Workers(Amazon EC2 t2.micro instances) with the variance value of 1, 2, 4, 8 and 16 with their execution time for for 10 ms, 1 sec and 10 sec of sleep operation. Total computed tasks = Time * No. of instances(Workers).

  Efficiency= Ideal Time (10ms, 1sec, 10 sec) * 100/ Measured Time (sec) %
- The variation in efficiency for 1 worker is shown with 10ms, 1 sec and 10sec is very high, while for 2, 4, and 8 workers the variation is very less for 10ms, 1 sec and 10 sec.

- **Local Client-worker :**
  - **Throughput:**



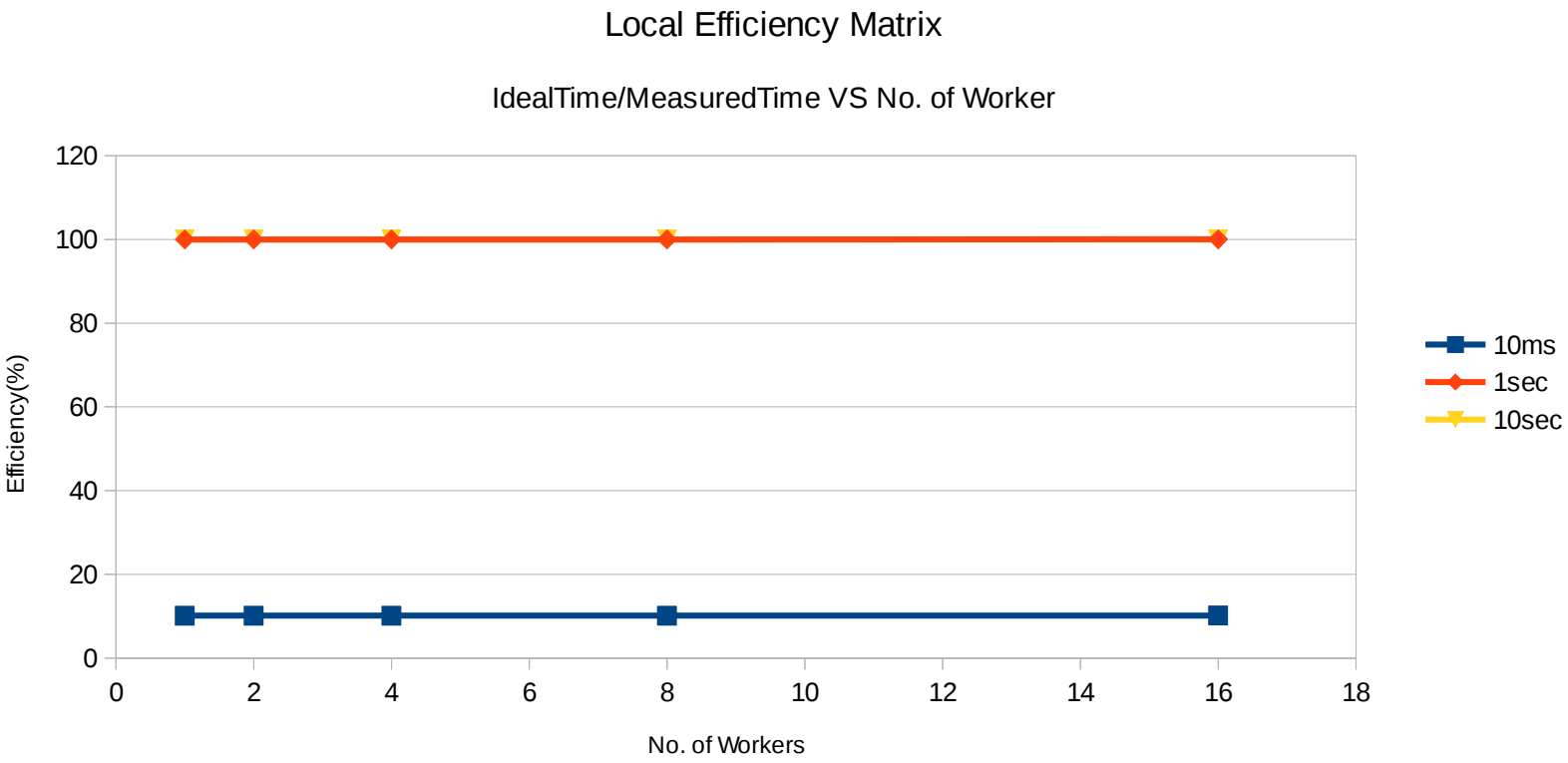
No. Of Worker	Throughput	Measured Time	Tasks
1	249850.089946032	10.006	2500000
2	166466.906379012	15.018	2500000
4	176815.899285664	14.139	2500000
8	178673.527730132	13.992	2500000
16	166190.25460347	15.043	2500000

- The above graph shows the throughput calculation of Local through 100k Sleeep(0) operations. The graph is drawn for the variance of No. of Workers(threads) with the variance value of 1, 2, 4, 8 and 16 with their throughput value.

  Throughput= Tasks / Measured Time

- From the graph it is shown that throughput decreases as the number of the workers doubled. Reason for this is that cpu cores in local processors are fixed and less. So, as no. of threads for executing the task increases the fixed no. of task gets chance to be executed, while remaining has to wait. So, the throughput value decreases.

- **Efficiency:**



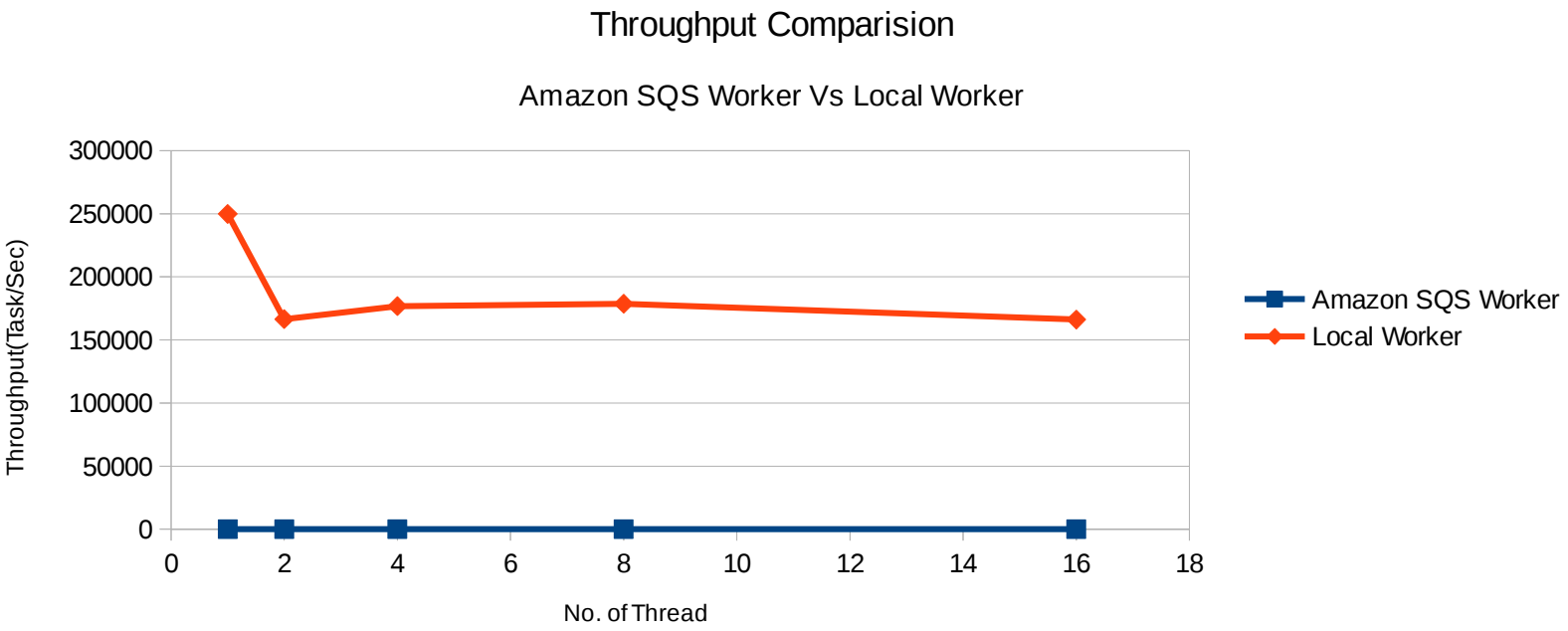
No. Of Worker	10ms	1sec	10sec
1	98.4058256249	99.9800039992	99.99800004
2	98.4930562395	99.9800039992	99.99700009
4	98.4445757039	99.9810036093	99.99700009
8	98.590160702	99.9820032394	99.99700009
16	98.0584428319	99.9780048389	99.99500025

- The above graph shows the efficiency calculation of Local workers. The graph is drawn for the variance of No. of Workers(threads) with the variance value of 1, 2, 4, 8 and 16 with their execution time for for 10 ms, 1 sec and 10 sec of sleep operation. Total computed tasks = Time * No. of instances(Workers).

  Efficiency= Ideal Time (10ms, 1sec, 10 sec) * 100/ Measured Time (sec) %

- The variation in efficiency is negligible because the time to execute the task is nearly similar to the ideal time needed to execute the task.

- **Throughput Comparision :Amazon SQS Worker VS Local Worker :**



No. Of Worker	Amazon SQS Worker	Local Worker
1	13.15	249850.089946032
2	13.26	166466.906379012
4	13.57	176815.899285664
8	13.85	178673.527730132
16	14.14	166190.25460347

- The above graph shows throughput comparison of Amazon SQS worker and Local Worker(Theads). It is clearly visible that for fixed no. of workers the Amazon SQS Wokers are having very less variance for 1, 2, 4, 8, and 16 instances while the Lacal Workers are having major variance from 1 to 2 and after 2, for 4, 8 and 16 the variation in the throughput is very less.