University at Buffalo
The State University of New York

PARALLEL COMPUTING

K-MEANS CLUSTERING

USING MPI

INSTRUCTOR : DR. RUSS MILLER

PRESENTED BY : NEEL DUNGARANI

CONTENT

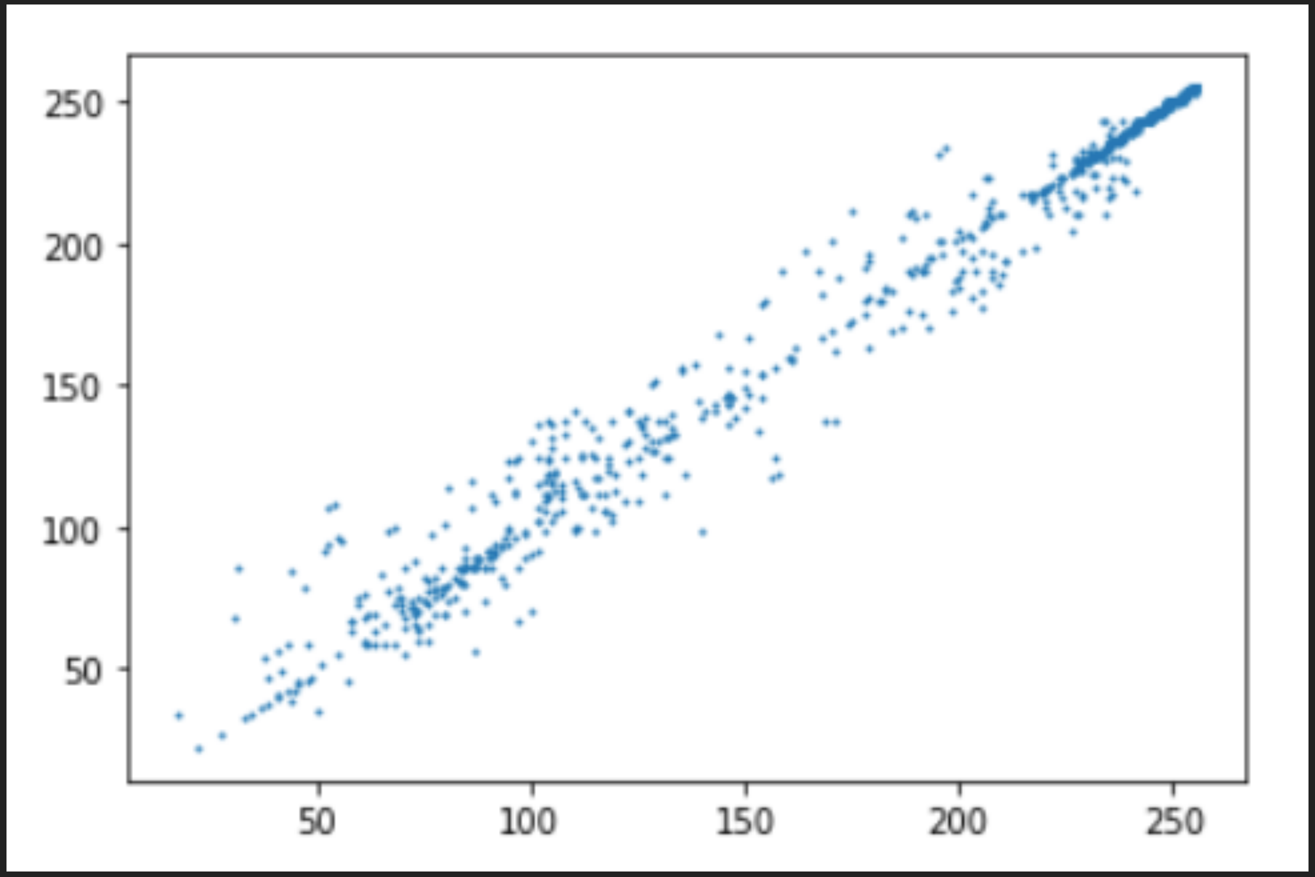
- K-means
- Sequential algorithm
- Parallel Model
- Performance readings
- Observation

K-MEANS

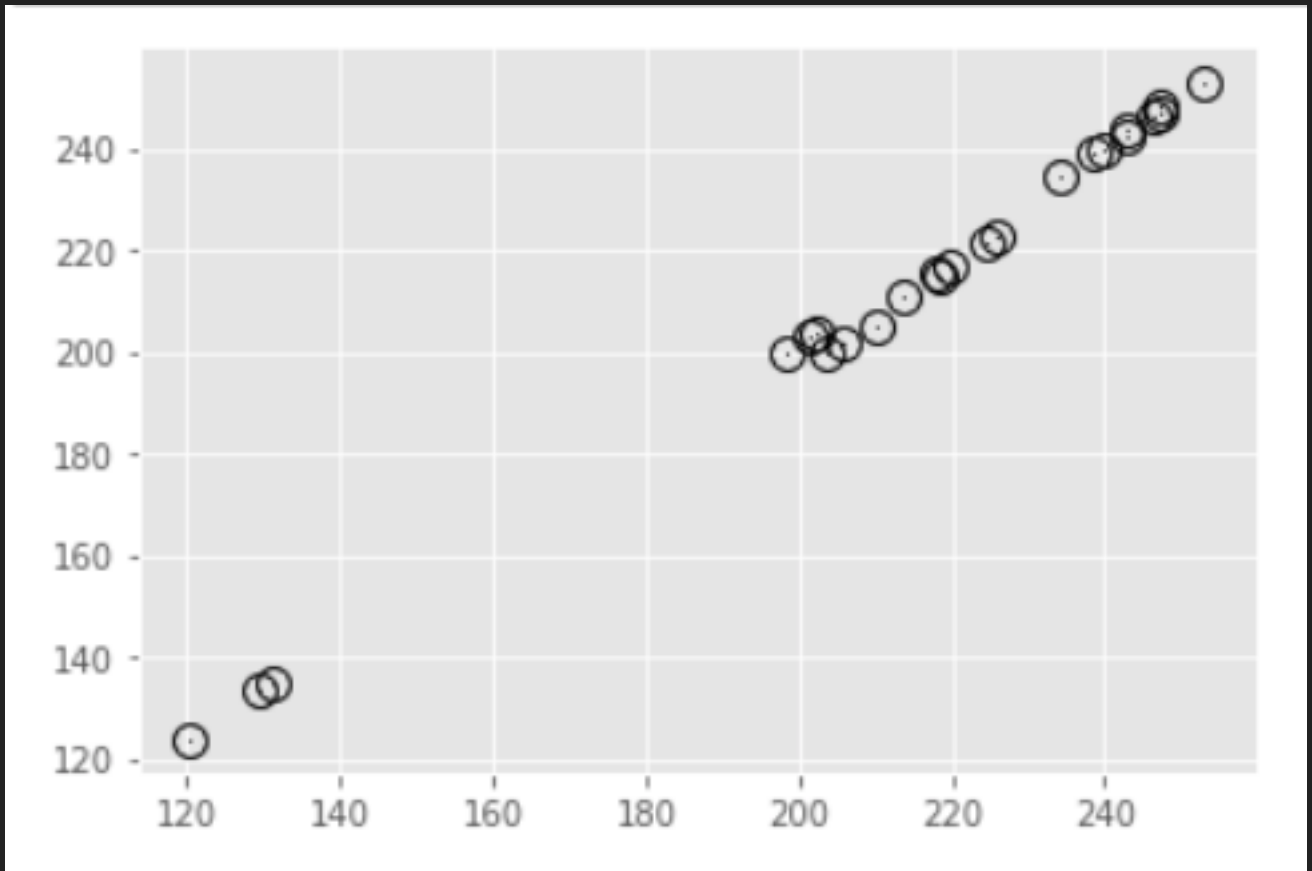
K-MEANS ALGORITHM

1. Pick K points as initial centroids from the data set, either randomly or the first K.
2. Find the euclidean distance of each point in the data set with the identified K points - cluster centroids.
3. Assign each data point to the closest centroid using the distance found in the previous step.
4. Find the new centroid by taking average of the points in each cluster group.
5. Repeat 2 to 4 for a fixed number of iteration or till the centroids don't change.

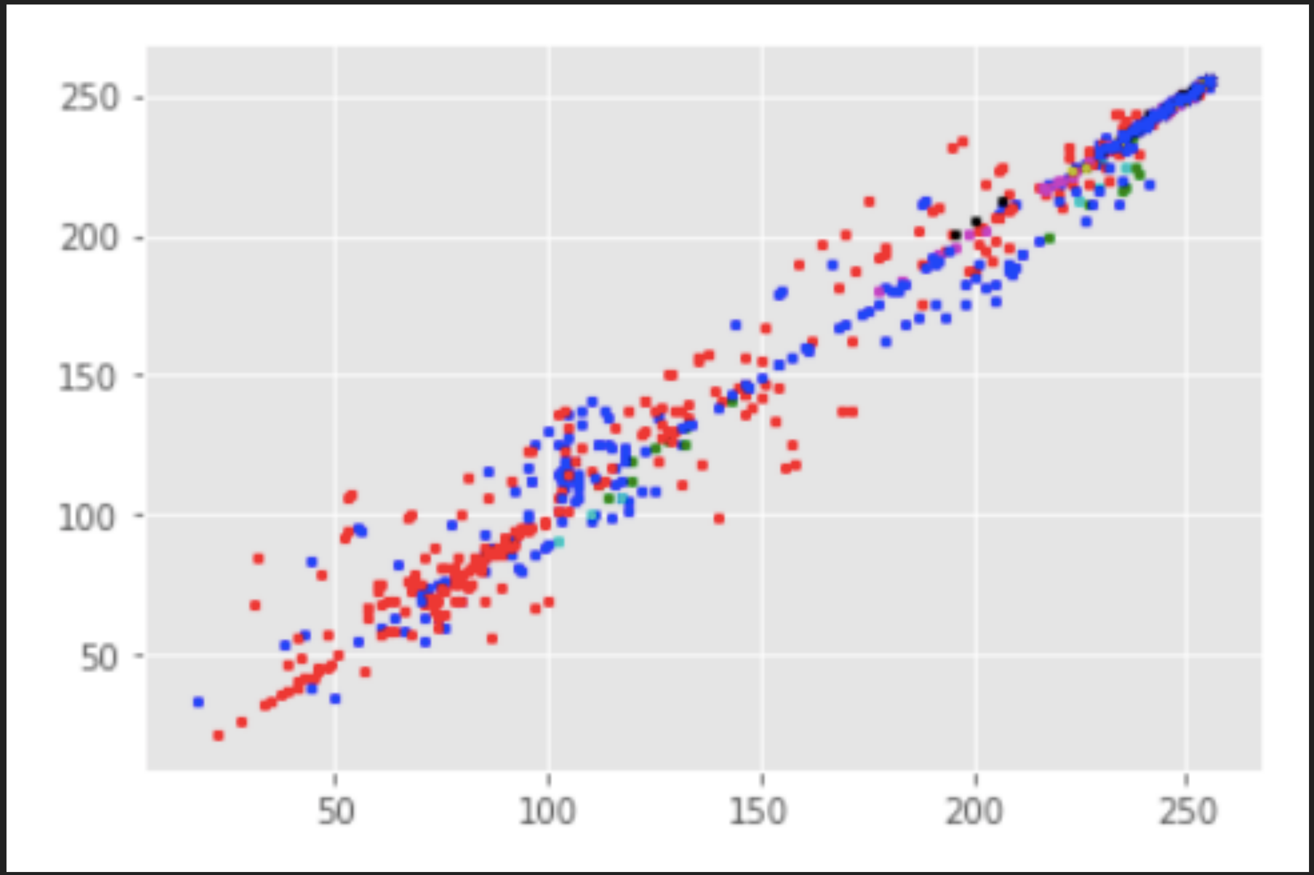
SEQUENTIAL IMPLEMENTATION



Cluster Centroids



Data clusters



K = 8 clusters

PARALLEL IMPLEMENTATION

PARALLEL APPROACH

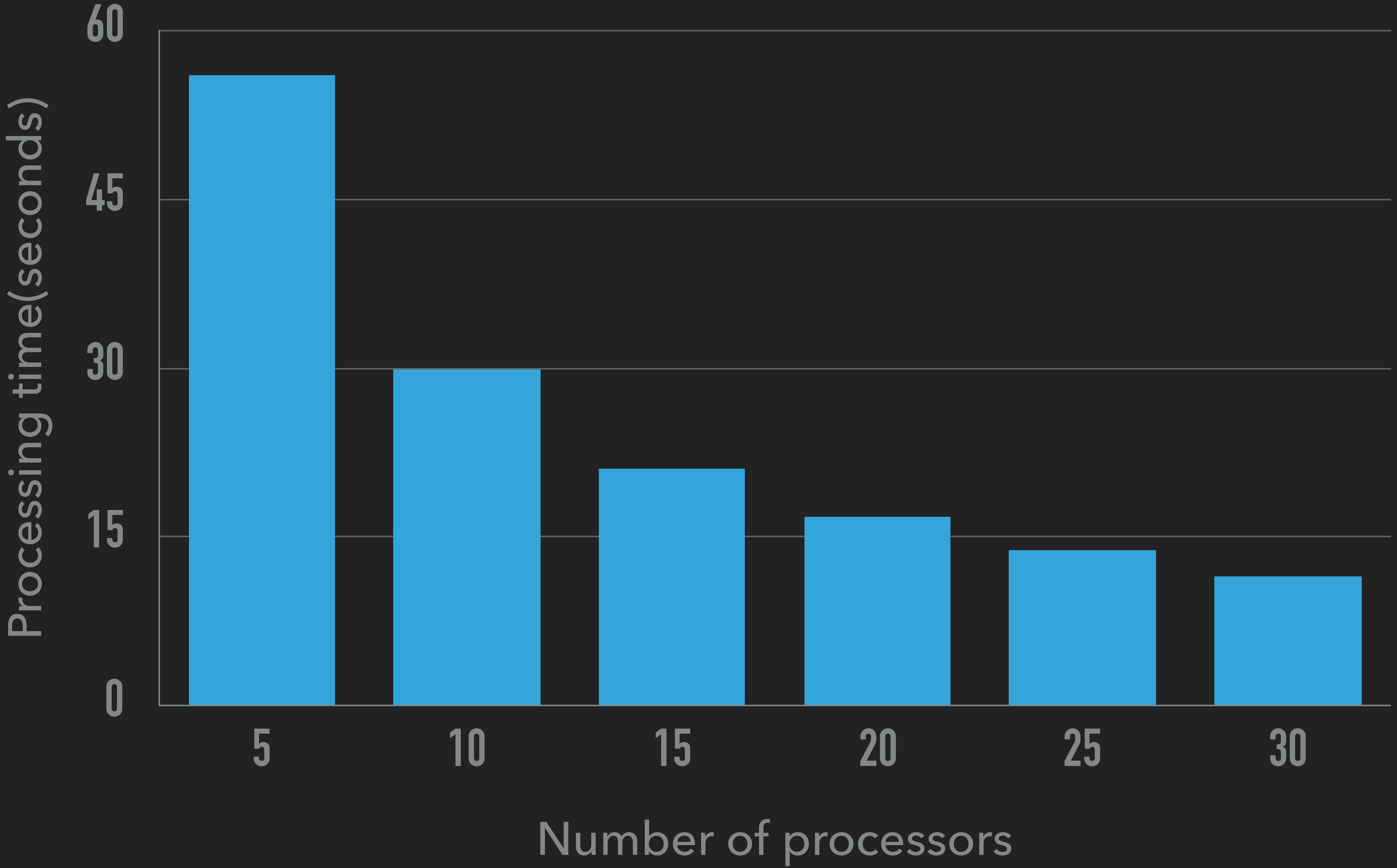
- Divide the training set evenly into multiple sets for each processor
- Initially take random data-points as centroids and broadcast them to each processor
- Calculate minimum euclidean distant centroid for each point on processor and add them to respective clusters
- Calculate sum of all data in each clusters and send final sum and number of data to processor 0.
- On processor 0 gather all sum and length to calculate new centroids.
- Broadcast the new centroids to each processor and repeat the above process for fixed number of times.

PERFORMANCE READINGS

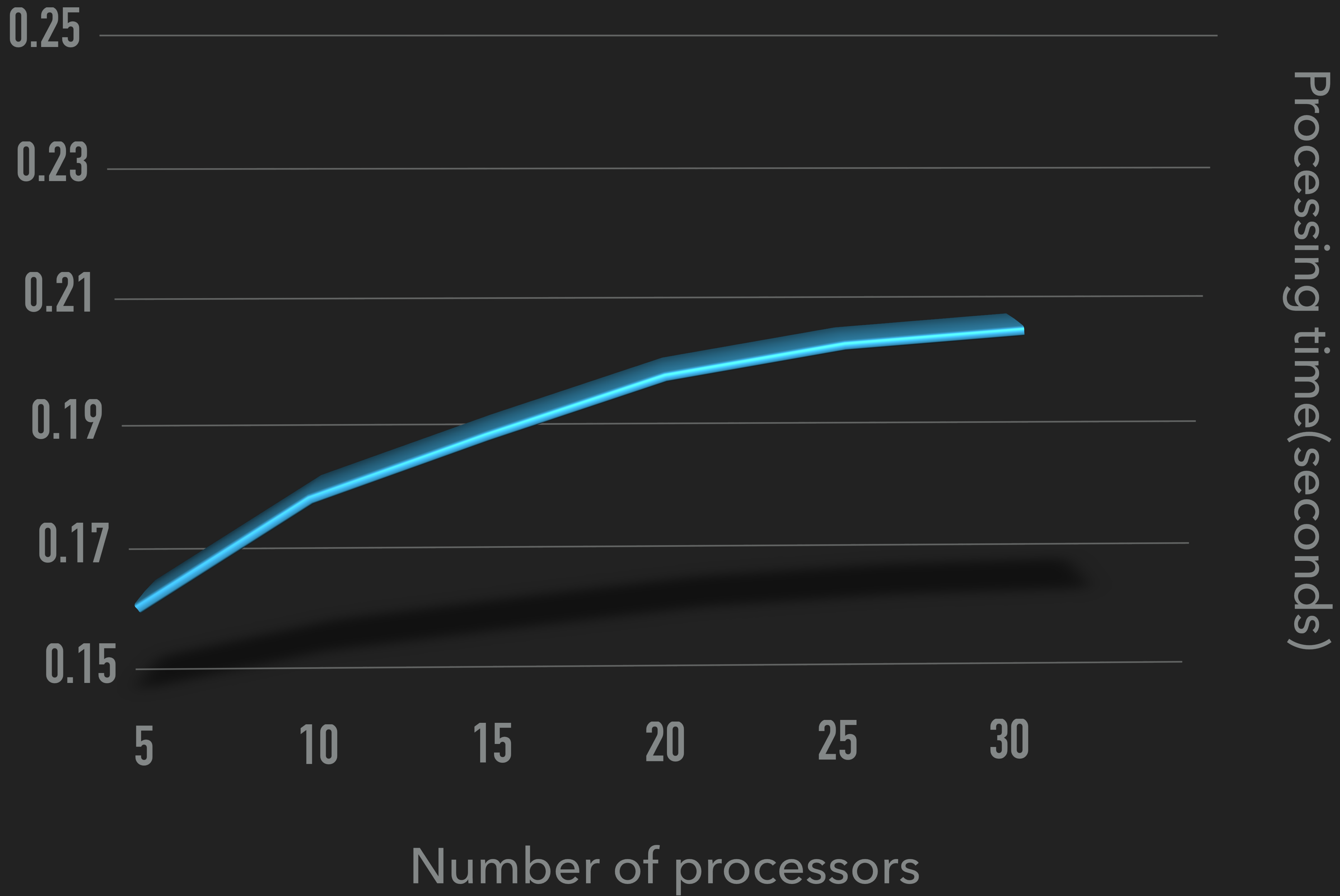
AMDAHL'S SPEED-UP

	5	10	15	20	25	30
1000	0.008187389374	0.005753993988	0.005330371857	0.005474233627	0.005343437195	0.005710554123
10000	0.06578499079	0.03853297234	0.03117996454	0.02483195066	0.022611022	0.02051025629
100000	0.6585662689	0.3534529743	0.2510416508	0.2010882378	0.165844202	0.1403254509
1000000	6.607971458	3.50974481	2.511008978	1.97383976	1.653262913	1.389199734

AMDAHL'S GRAPH FOR 4K-IMAGE (8294400 DATA POINTS)

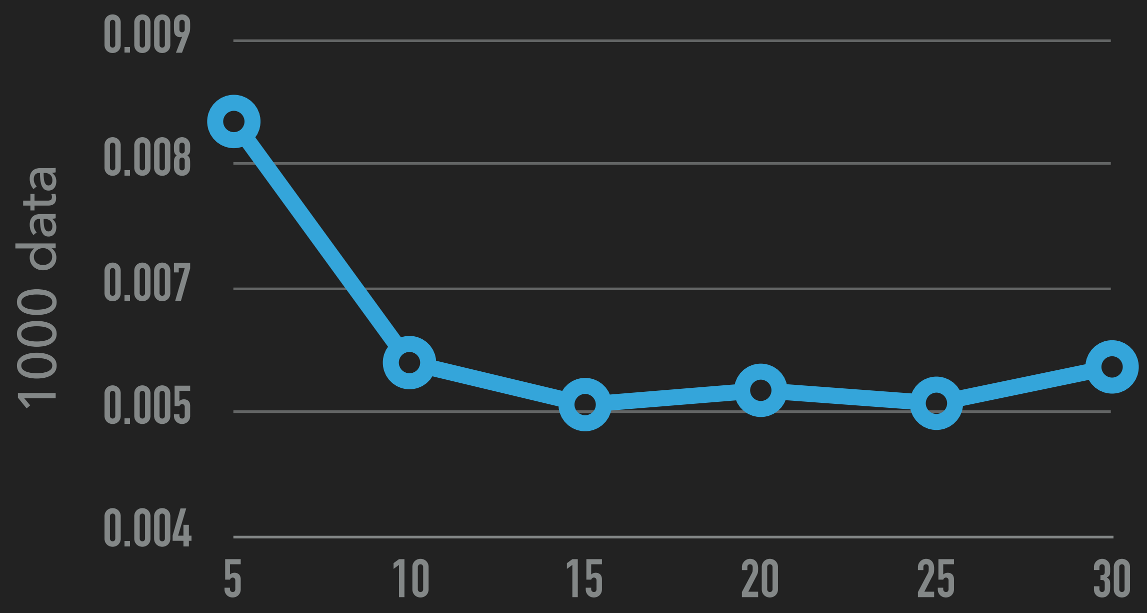


GUSTAFSON'S GRAPH FOR 5000 DATA-POINTS PER PROCESSOR



OBSERVATION

- Speedup initially starts with ~90% and later it decreases and there is minimal difference as we increase processors from 25 to 30.
- With less number of data this algorithm does not seem useful as increase in speed is less and it also gives strange readings.



- Looking at all readings we can say that this algorithm performs best around 20-25 number of processors.

THANK YOU..!!

NEEL DUNGARANI