Northeastern
Khoury College of Computer
and Information Sciences

CS 6240 Final Project Presentation

Chin Shiang Jin

Investigate efficient way(s) to compute the optimal waiting location for the drivers based on the Uber pickups in New York City Dataset using the K-means clustering algorithm.

Northeastern
Khoury College of Computer
and Information Sciences

Project Overview

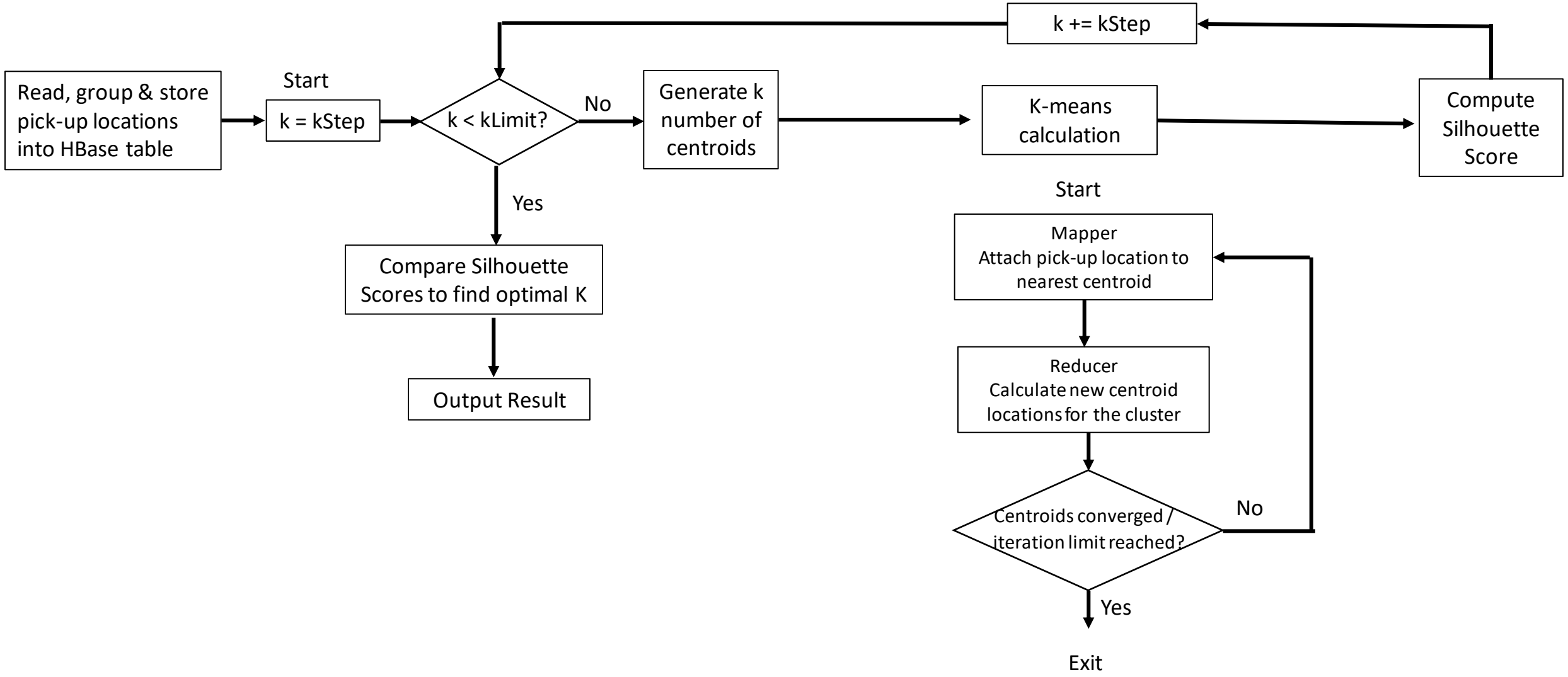
- **Project Goal** : Compute the optimal waiting locations for Uber drivers based on pick-up locations history in New York City using K-means clustering algorithm run on Map-Reduce framework

- **Evaluation :** the optimal K values is evaluated using Silhouette Score

- **Data :** Coordinate of the pick-up locations stored in csv file, where the Latitude and Longitude of the pick-up location can be extracted

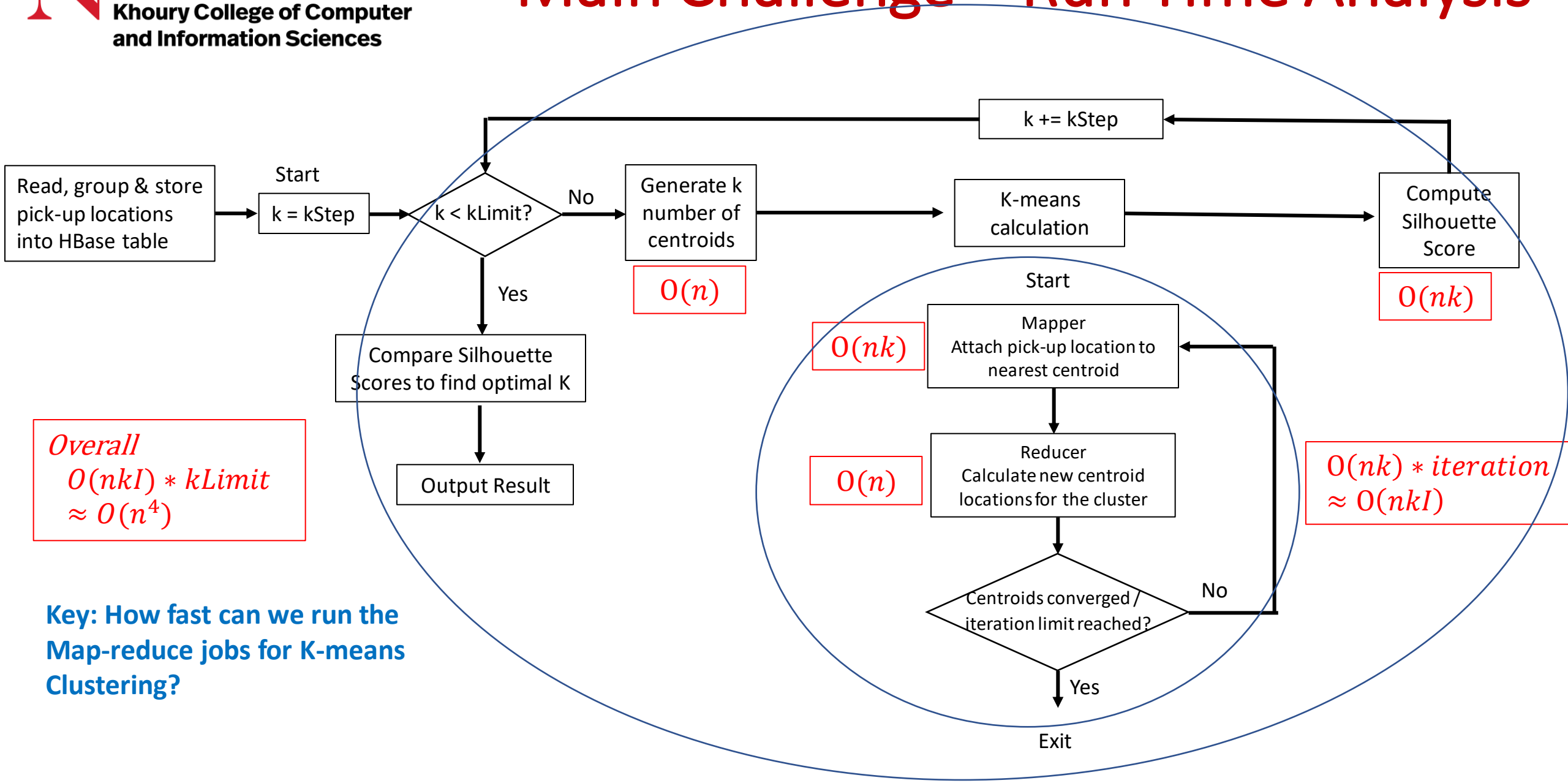
Northeastern
Khoury College of Computer and Information Sciences

Main Program Design



Northeastern
Khoury College of Computer and Information Sciences

Main Challenge – Run Time Analysis



Overall
$O(nkI) * kLimit$
$\approx O(n^4)$

$O(nk) * iteration$
$\approx O(nkI)$

Key: How fast can we run the Map-reduce jobs for K-means Clustering?

Northeastern
Khoury College of Computer
and Information Sciences

Solving Main Challenge

- **Goal** : Minimize run time for a map-reduce job to compute K-means clustering algorithm
- **Potential Bottlenecks :** Computations time, Jobs set up overhead, IO overhead
- **Result :** (Computations time unlikely to be the bottleneck)

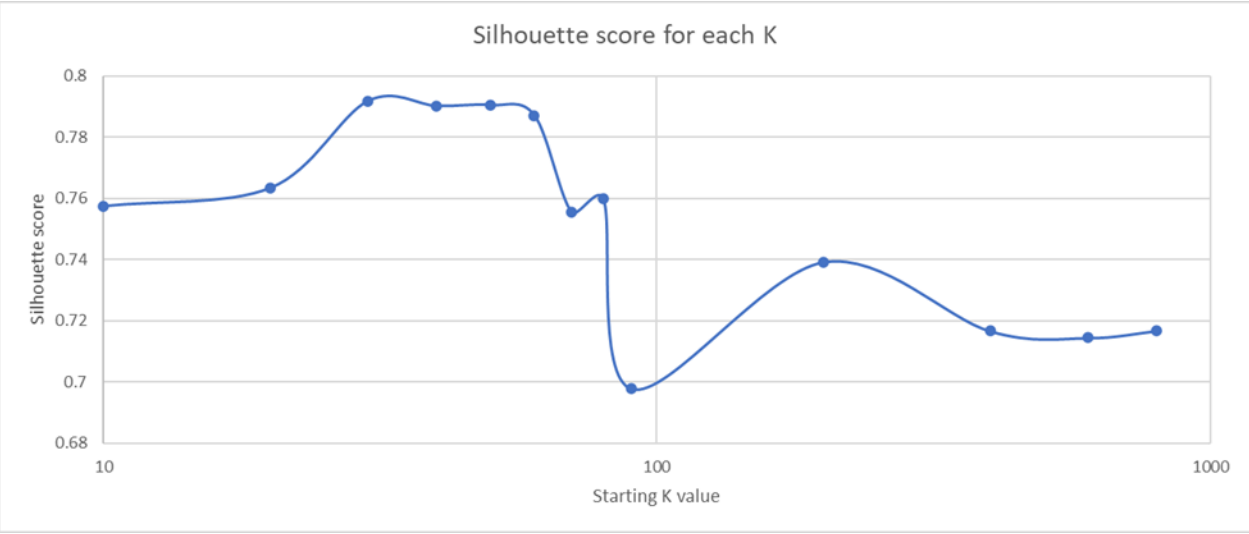
MR job Description	Main Run	4 x k	4 x n
Run time for each MR	38 – 40 seconds	38 – 40 seconds	38 – 40 seconds

- **IO overhead :** Currently writing data into HBase memory and reading data from it, could it be faster?
- **MR jobs set up overhead :** likely the culprit

Northeastern
Khoury College of Computer and Information Sciences

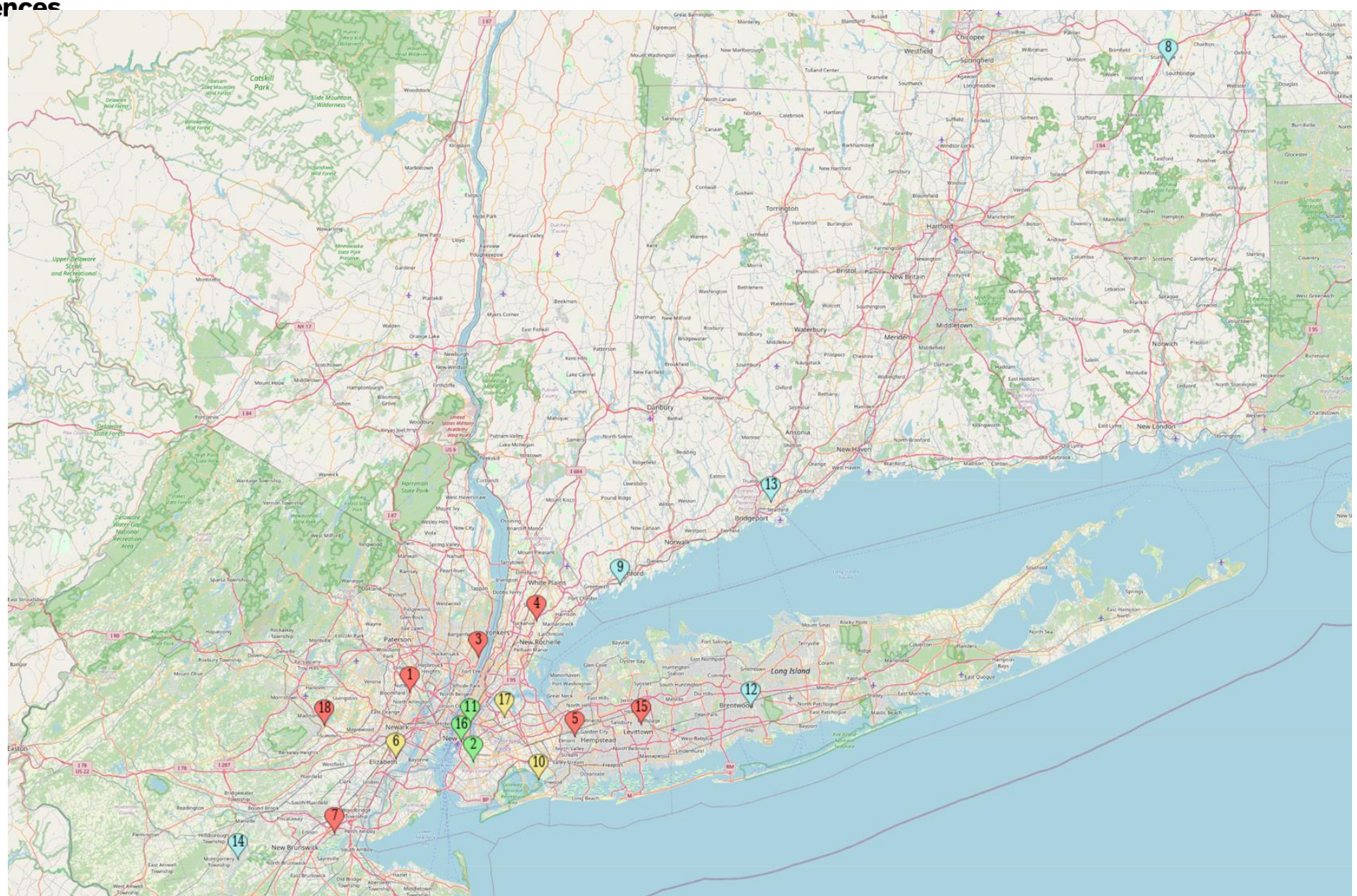
More Result

Computation Details	Machines Details	Total Run Time
22,000 pick-up locations K value from 10 to 100 in the step of 10, maximum 20 iterations	2 machines, of m5.xlarge	2 hours 34 minutes
22,000 pick-up locations K value from 10 to 100 in the step of 10, maximum 20 iterations	10 machines, of m5.xlarge	1 hour 58 minutes
22,000 pick-up locations K value from 200 to 1000 in the step of 200, maximum 20 iterations	10 machines, of m5.xlarge	54 minutes, 58 seconds
86,117 pick-up locations K value from 10 to 100 in the step of 10, maximum 20 iterations	10 machines, of m5.xlarge	2 hours, 1 minute



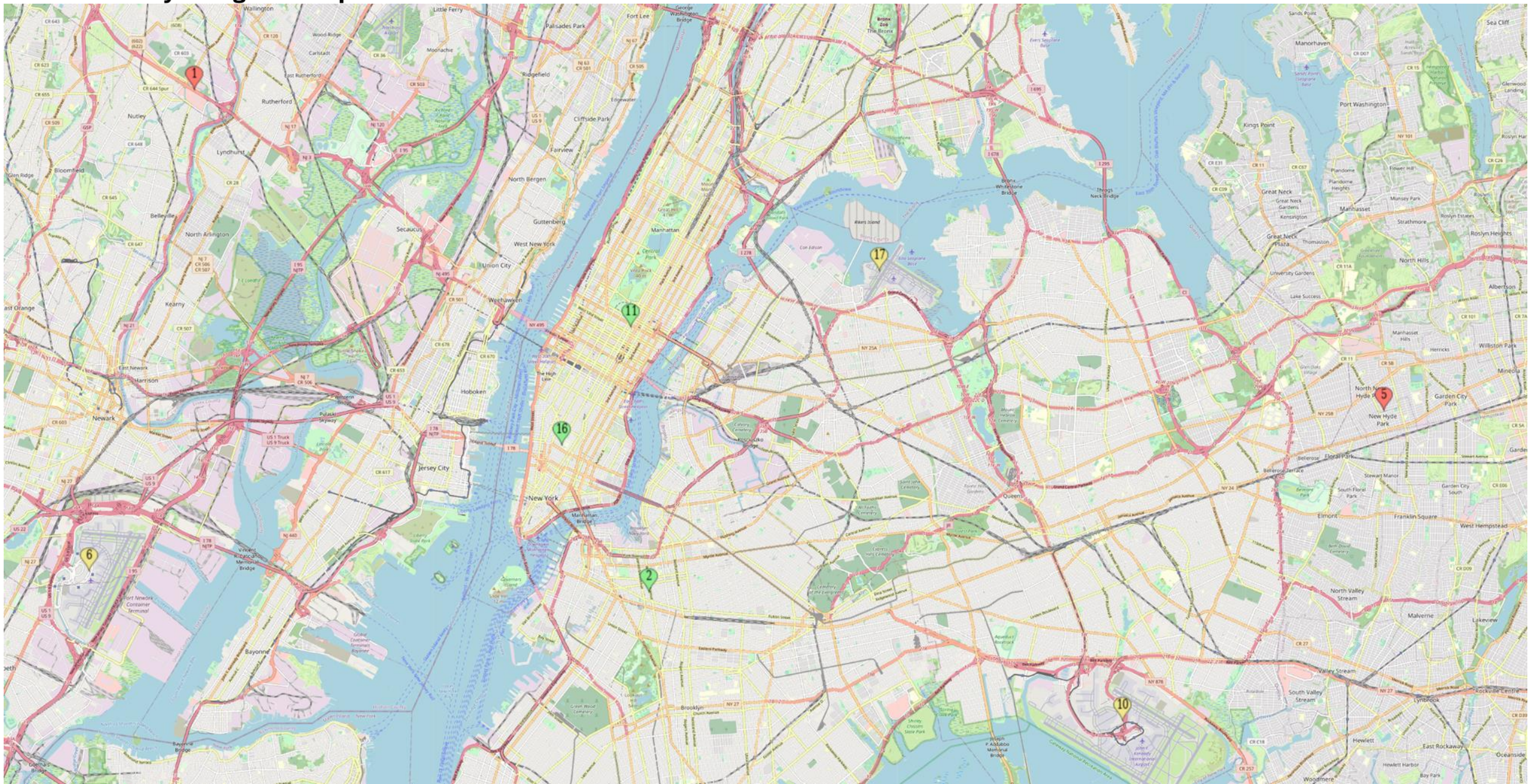
Northeastern
Khoury College of Computer and Information Sciences

Optimal Waiting Locations



Northeastern
Khoury College of Computer

Optimal Waiting Locations



Northeastern
Khoury College of Computer
and Information Sciences

Conclusion

- Current program can compute the optimal waiting locations but have limited application in Real-time.
- Still can be used if the result is not required immediately
- Similar map-reduce programs will likely have same bottleneck : map-reduce set up overheads

Northeastern
Khoury College of Computer
and Information Sciences

Q & A

Thank You!