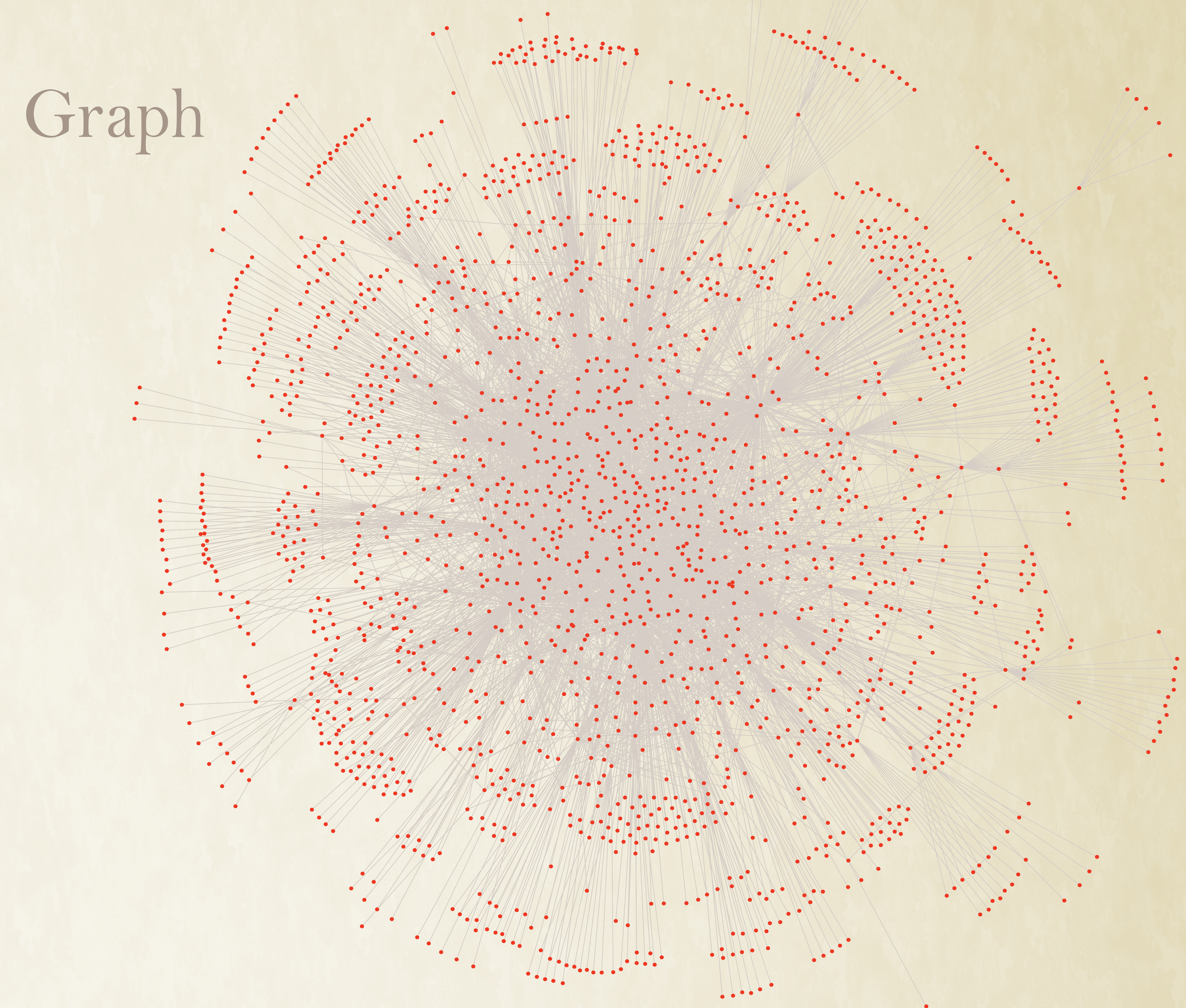
GRAPH ALGORITHM – EFFICIENT SHORTEST PATH ESTIMATION

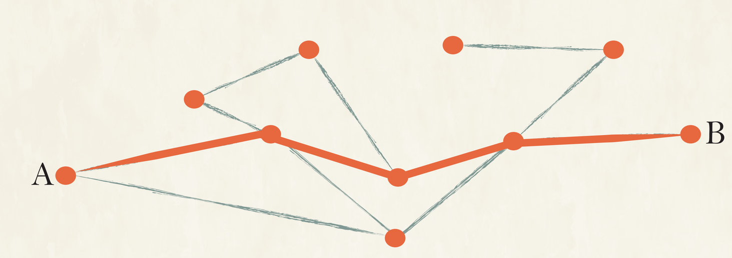
Yonk Shi[1], Arijit Khan[2], Xifeng Yan[2]
1. Moorpark College
2. University of CAlifornia, Santa Barbara, Department of Computer Science



F-1: A real computer graph of Last.fm database

INTRODUCTION

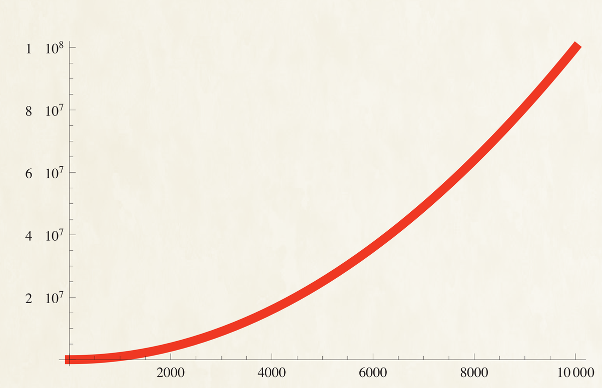
In computer science, a graph is a network of data (F-1). Unlike a database, data is stored as a network. It is the building block for many of your favorite services like Google, Facebook and Pandora. For example, Pandora uses graphs to store all the relationships between songs, and that is how it can recommand similiar songs for the user. Such network is composed of many algorithms. In our specific research project, we focus on the shortest path algorithm for massive graphs. (F-2) shows a very simple shortest path algorithm. Shortest path is often part of a bigger and more complex algorithm such as ranking, relationship analysis etc. Therefore our goal is to design a general purpose shortest path that can be adapted by more complex algorithms.



F-2: A simple shortest path algorithm illustrated

CURRENT ALGORITHMS

Dijkstra's classic algorithm operates based on a very simple principle: discover every single possible route. However, if we are to calculate a shortest path of a very large graph (e.g. Facebook) not even a super computer could handle it. (F-3) Shows the time it would take as graph gets larger.



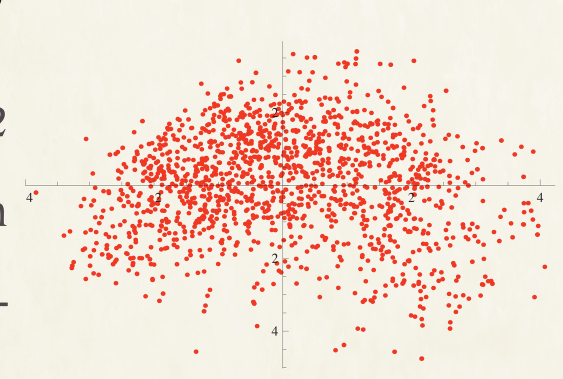
F-3: Time complexity graph of Dijsktra's algorithm

MULTI DIMENSIONAL SCALING

We decided to take a look at the real world by taking a look at what humans do when looking for a shortest path on a map. We used a process called MultiDimensional Scaling and implemented it using Wolfram Mathematica:

$$q_{ij} = \frac{\exp(-\|y_i - y_j\|^2)}{\sum_{k \neq i} \exp(-\|y_i - y_k\|^2)}.$$

This process has the ability to convert abstract data into a 2 dimensional map(F-4). For example, it can assign each friend of yours in your friends network a x and y coordinates, and you can use those coordinates to lay them out on a piece of paper.



F-4: Last.fm data converted into MDS

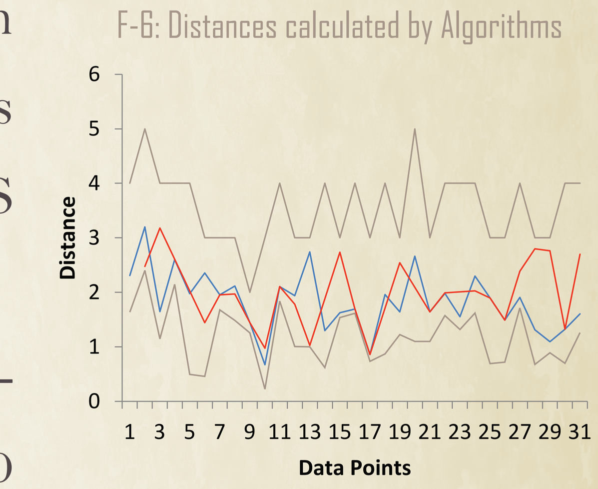
OUR ALGORITHM

Based on the results of MDS, we have designed a greedy algorithm which uses a statistical priority model that can "crawl" towards destination in a two dimensional space (F-5).

F-5: A shortest path algorithm calculated using MDS

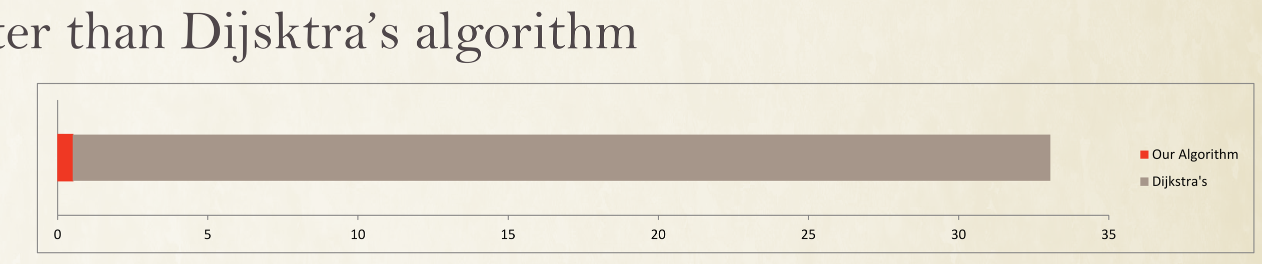
RESULTS

We have benchmarked our algorithm with Dijkstra's algorithm(F-6). Our algorithm (red) is very close dijkstra's algorithm mapped in MDS (blue) The average difference is 150%.



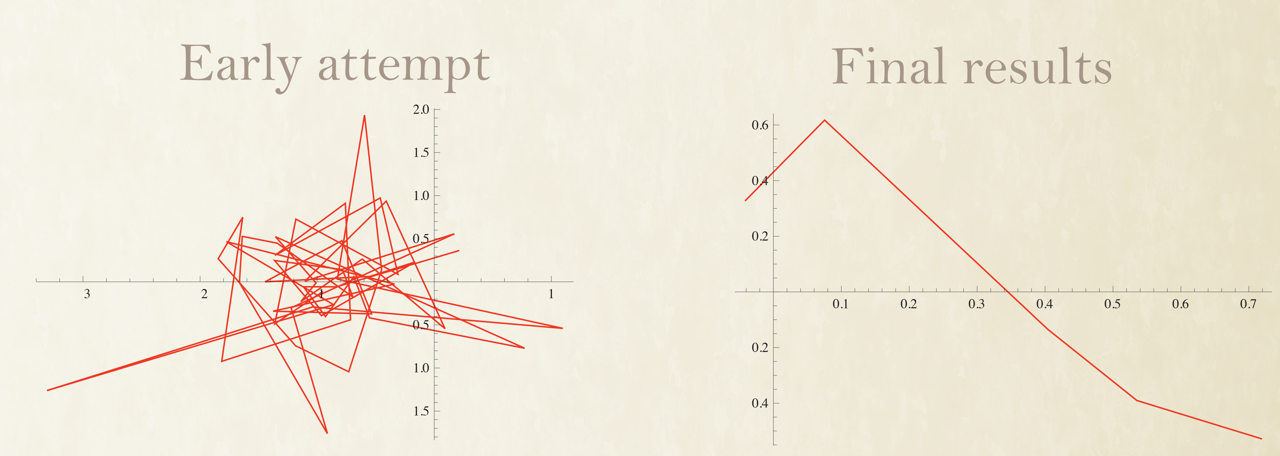
F-6: Distances calculated by Algorithms

(F-7) shows the time (in seconds) taken by different algorithms. Our algorithm is as high as 3000 times faster than Dijsktra's algorithm



F-7: Average time taken by algorithms in seconds

We have come a long way to achieve such high accuracy. (F-8) demonstrates an early version of our algorithm as compared to our most recent falogirthm.



F-8: A failed result vs a successful result

We have set up many limiting parameters in order to achieve such high acuracy but consequently there are 10% failure rate at which no results were

FUTURE GOALS

We are constantly working on better and faster algorithms. Our primary goal right now is to eliminate the 10% failure rate and improve efficiency. We are also planning on implementing "tagging" information to homogeneus data. Tagging information allows us to reduce massive graphs thus further improve our accuracy and efficiency.

ACKNOWLEDGEMENT

I thank the staff of INSET for providing such an amazing opportunity for me to work at UCSB. I thank UCSB and Xifeng Yan. I thank my mentor Arijit Khan for providing guidance throughout my research.

UCSB

INSET
Internships in Nanosystems Science, Engineering and Technology

CSEP
Center for Science and Engineering partnerships

REFERENCE:

Joshua B. Tenenbaum,1* Vin de Silva,2 John C. Langford3, A Global Geometric Framework for Nonlinear Dimensionality Reduction, 2000
Lawrence K. Saul, Sam T. Roweis, An Introduction to Locally Linear Embedding, 2000