The k-Median Problem: A Literature Review

Research Project

Samar G. Sajnani

Internet Algorithmics (CS4438A)

Department of Computer Sceince

Western University

11/25/2018

Course Instructor: Dr. Roberto Solis-Oba

1. Abstract

The k-median problem is one of a few NP-hard problems, in fact, the generalized version of the problem is seldom investigated because of its' lack of approximation. Instead the metric k-median problem is the more prominent version that is studied and has produced tighter and tighter approximation ratios. Each new study of the problem is trying to find new ingenious ways to approximate the problem. There have been a few papers over the past few decades that have shown consistent progress towards 3-approximation ratio algorithms. These studies include techniques such as primal-dual linear programs, bi-point solutions, convex combinations, filtering, consolidation, dependent rounding, location search, and swaps. The solutions of this problem are characterized by heuristics, linear programs and location searches. For location search the lowest approximation ratio of $3 + \frac{2}{p}$ was found by Arya et al., in 2001, using a method of swapping called p-swap. The most recent discovery in the development of linear program solutions was made by Li and Svensson in 2012, when they discovered a $1 + \sqrt{3} + \varepsilon$ ratio approximation algorithm with time complexity of $O(n^{O(1/\varepsilon^2)})$, where $\varepsilon > 0$. There is a need for lower approximation ratios because currently the cost of an approximate solution can be more than twice the cost of the optimum solution.

2. Introduction

Here we will introduce the k-median problem in a simple form and give an overview of its significance and the current solutions available for the problem.

2.1. Description of the k-Median Problem

Let us start with an example, imagine a large company is investing significant amounts of money to create a distribution network for their product. The company supplies their product to n number of stores where the product is sold. The requirements by the company are that they can afford to convert exactly k number of their stores into manufacturing facilities. The delivery costs for their product is completely dependent on the distance from the manufacturing facilities to the stores. Each store will be supplied by exactly one manufacturing facility. As such, the company would like to minimize its cost of delivery to its stores. Which stores should be selected to become manufacturing facilities such that the overall cost of delivery is minimized? This is a description of the k-median problem, and this example portrays the high industrial relevance of the k-median problem (Kuehn & Hamburger, 1963). Of course, this is not solely a problem of industrial relevance, it is generalized as an optimization problem for networks.

2.2. Significance of the k-Median Problem

Using the above example, the stores can be replaced with data-points; framing the problem as a data mining and clustering problem (Bradley et al., 1998; Mulvey & Crowder, 1979). Additionally, the stores can also be replaced by internet service providers exemplifying an internet distribution service (Andrews & Zhang, 1998). These are a few of the many real-world applications for the k-median problem, as a result, it is important to find an efficient solution to this problem.

Finding a solution to the k-median problem is not trivial because of the complexity class that the k-median problem belongs to. There have been no polynomial time solutions for this problem because of its combinatorial nature. In finite math, the k chosen stores are identified through a combination C(n,k), where n is the number of stores and the problem is defined as n choose k. Combinations involve the use of factorials, as such deriving a solution for the problem is non-polynomial in time because factorials are non-polynomial. However, verifying a k-median solution is a polynomial time task that involves looking for a lower delivery cost than the one proposed by the solution, if one does not exist then the solution is optimal. The non-polynomial nature of the solution and the polynomial nature of its verification place this problem in the non-deterministic polynomial (NP) class of problems. Specifically, the problem is part of the NP-hard set because a polynomial time solution for the problem could be used to derive polynomial time solutions for all other NP problems.

Due to the non-trivial nature of the k-median problem and its importance in the real-world, an efficient solution to the k-median problem is highly sought after.

2.3. Facility Location Problem

The k-median problem is a special case of the facility location problem; we will describe the facility location problem as an LP problem and derive an LP definition for the k-median problem. Let's start with facility locations; if we locate a facility i there is a fixed cost of locating the facility known as f_j, there is capacity for the facility C_j which is the maximum production for the facility. If a facility is chosen, then its corresponding value Y_j is equal to one, else it is zero. A client i has demand D_i that must be met and must be connected to at least one facility. Products are manufactured in the facilities and there is a cost for transport between the facilities and the clients known as c_{ij}. The quantity transported from a facility j to a client i is represented by X_{ij}. Sets CL and F represent the clients (i) and facilities (j), respectively. Develop a distribution network that chooses the optimal number of facilities to minimize the total cost of transportation, meet the production capacity and the demand of each of the clients.

(1)

$$Minimize \sum_{j \in F} f_j Y_j + \sum_{j \in F} \sum_{i \in CL} c_{ij} X_{ij}$$

Equation (1) is the main condition of the LP problem for the facility location problem. The first sum of the minimize function will only hold for facilities that have been established because Y_j is one and the cost of locating that facility will be added to the cost function as f_j. We cannot complete our formulation of the facility location problem without adding the appropriate constraints to the minimize function.

(2)

$$\forall j \in F: \sum_{i \in CL} X_{ij} \leq C_j \mathrm{Y}_j$$

Plainly, the first constraint (2) states that the sum of all products transported from a facility to its clients cannot be greater than the product produced at the facility, i.e. the capacity of the facility.

(3)

$$\forall i \in CL: \sum_{j \in F} X_{ij} \geq D_i$$

The second constraint (3) ensures that the sum of product transported to a specific client is not less than the demand of that client. The last two constraints are $Y_j = 0,1$ and $X_{ij} \geq 0$; the first stating that an entity is either a client or a facility, the second ensuring that the quantity of product transported to an entity is not negative. Resulting from this minimization and its constraints is the facility location problem stated as a linear program, it is a linear program that is general and encompasses many different cases. One of these special cases manifests as the k-median problem, specifically the case where the cost of locating a facility (f_j) is one, the cost of transport from a facility j to a client i (c_{ij}) is one, the demand D_i of each client i is one, and the quantity transported from a facility j to a client i (X_{ij}) is zero or one. With only these changes there is no restriction on the number of facilities. Thus, a new constraint, described by equation (4), limits the number of facilities to k.

(4)

$$\sum_{j \in F} Y_j = k$$

Additionally, the constraint in equation (2) needs to be updated to ensure that each client is only supplied by one facility. This requires replacing equation (2) with the following constraint:

(5)

$$\forall j \in F \; \forall i \in CL: \quad X_{ij} \leq Y_j$$

This described form represents the k-median problem as a case of the facility location problem.

2.4. Proposed Solutions for the k-Median Problem

Since no polynomial time algorithms have been found to solve the k-median problem and it is likely that no polynomial time algorithm exists, most fruitful efforts to find a solution have been in the research of approximation algorithms. As the name suggests, approximation

algorithms produce solutions that are close to the optimal solution in polynomial time, however, they do not ensure that the optimal solution is found. Approximation algorithms are often used to approximate solutions for NP problems, for polynomial (P) problems an approximation is not needed. To approximate a k-median solution, two main types of approximation algorithms have been used: a local search method and a linear programming (LP) approach.

Before we discuss approximation algorithms, it is important to review a couple of concepts. The first concept is that of approximation ratio, which is calculated using cost functions and for a minimization algorithm is defined by the cost of the approximate solution over the cost of the optimal solution. For a maximization algorithm, the approximation ratio is the cost of the optimal solution over the cost of the approximate solution. As such, the closer the approximation ratio is to one the closer the approximate solution is to the optimal solution. Second, the hardness of an approximation algorithm is defined as a hard-lower limit for the approximation ratio of an NP problem. Lastly, if an approximation ratio is tight this means it is the only approximation ratio for a specific approximation algorithm, it is both the upper and lower limit. Now that we know these concepts we can continue our study of the k-median approximation algorithms.

From a historical perspective, the k-median problem is NP-hard, the general problem cannot be approximated for any approximation ratio $c > 1$ unless P is NP. As a result, many researchers resort to a specific case of the k-median problem known as the metric k-median problem. For the metric k-median problem, the cost function from one client to another is symmetric, essentially c_{ij} is equivalent to c_{ji} for any two clients. Additionally, the triangle inequality holds for the distances between the clients given by $c_{ik} \leq c_{ij} + c_{jk}$ for any three clients in the network.

There has been a greater amount of success in the development of approximate solutions for the metric k-median problem. The LP approach was demonstrated in December of 1992, it was shown that approximations for geometric medians could be established using LP and the first approximation algorithm for the metric k-median problem was developed. The approximation algorithm had a ratio of $1 + \epsilon$ and in response to a k value that could select up to $(1 + \frac{1}{\epsilon})(\ln(n) + 1)$k centers, another algorithm was also presented with ratio $2(1 + \epsilon)$ if it is allowed to select up to $(1 + \frac{1}{\epsilon})k$ centers (Lin & Vitter, 1992). Bartal developed the first approximation algorithm for the metric k-median problem based on LP for at most k centers, in 1996. This method used trees to represent metric space and had a running time of $O(lognloglogn)$ (Bartal, 1996). The LP method was further investigated in July of 1999 and a constant approximation algorithm which used a dependent rounding technique was derived with a $6\frac{2}{3}$ approximation ratio (Charikar et al., 1999). In October of 1999, Jain and Vazirani improved on this approximation ratio by applying the primal-dual LP technique to the metric facility location problem and deriving the metric k-median problem. The algorithm has an approximation ratio of 6 and a time complexity of $O(mlogm(L + \log(n))$, where n and m are the vertices and edges respectively and L is the number of bits needed to represent a connection cost (Jain & Vazirani, 1999).

The local search approach for approximating the metric k-median solution was proposed in July of 2001. For single swaps, the algorithm approximates the solution with a ratio of 5 and for p swaps the ratio is $3 + \frac{2}{p}$, with $O(n)$ and $O(n^p)$ running times respectively (Arya et al., 2001). Jain and Vazirani used bi-point solutions, where a LP program is formed from the convex combination of two pseudo-solutions, usually k+1 and k-1, which are used to approximate the solution for k. They achieved an approximation ratio of 6 for the metric k-median problem based on their ratio of 3 for the facility location problem (Jain & Vazirani, 1999). In a subsequent paper, the facility location ratio is improved to 2; the resulting k-median ratio is 4 and the running time is $O(n^3)$ (Jain et al., 2003). Additionally, this paper improves on the hard limit of $1 + \frac{1}{e}$ as the lowest approximation ratio for the metric k-median problem. The new hard limit is $1 + \frac{2}{e}$ and is known as the hardness of the metric k-median problem (Jain et al., 2003). After a few years in 2012, Li and Svensson demonstrated a pseudo-algorithm with an approximation ratio of $1 + \sqrt{3} + \epsilon$ where $\epsilon > 0$. Their algorithm was based on the bi-point relation by Jain and Vazirani, except that they used k + O(1) facilities and then removed the O(1) facilities to get an approximation of the metric k-median problem. The running time of this algorithm is of the order $O(n^{O(1/\varepsilon^2)})$. This is just a glimpse of metric k-median solutions, in the following sections of this paper, we will investigate the methods and results of these algorithms in more detail, to identify how these solutions were established. We will compare the different algorithms and show that if a low running time is a priority then it comes at the cost of a less accurate approximation algorithm. A higher running time can be used to better approximate a metric k-median solution.

3. Linear Programming Approach

We will start our discussion with a review of the first integer linear programming approximation algorithm for the metric k-median algorithm as proposed by Lin and Vitter (Lin & Vitter, 1992). Followed by an analysis of the first constant ratio approximation algorithm demonstrated in 1999 by Charikar, Guha, Tardos and Shmoys (Charikar et al., 1999). Additionally, we will cover the ingenious paper by Jain and Vazirani that put forth the use of a primal-dual scheme for solving the metric k-median problem (Jain & Vazirani, 1999). Finally, we will analyze the newest approximation algorithm proposed by Li and Svensson in 2012 that has broken the long-standing approximation ratio of three established by the p-swap location search algorithm (Li & Svensson, 2012).

3.1. K-median Integer Programming Problem

Before working intimately with the k-median problem we need to formulate the integer linear programming form of the k-median problem. Earlier we had done so by deriving the k-median problem from the facility location problem. In this section, we will formulate a more simplified common LP form of the k-median problem. This form has the demand of each client and capacity of each facility equal to one. Additionally, the number of facilities is approximated up to the value of k to soften the requirement of exactly k facilities.

(5)

$$Minimize \sum_{j \in F} \sum_{i \in CL} c_{ij} X_{ij}$$

$$constraints:$$

$$\forall i \in CL: \quad \sum_{j \in F} X_{ij} \geq 1$$

$$\sum_{j \in F} Y_j \leq k$$

$$\forall j \in F \; \forall i \in CL: \quad X_{ij} \leq Y_j$$

$$\forall j \in F \; \forall i \in CL: \quad X_{ij}, Y_j \in \{0,1\}$$

We will use this construction in our discussion of the k-median problem approximation solutions proposed by Lin and Vitter. Particularly, a LP relaxation is required to soften the constraints to allow for real-number solutions, a real-number approximation is continuous as opposed to discrete and easier to solve using LP. To create an LP relaxation of the above formulation, the last constraint for which $X_{ij}, and\ Y_j$ are either zero or one becomes $X_{ij}, and\ Y_j$ are bigger than or equal to zero. The LP relaxation can then be optimized to generate a solution of $\hat{y}$ facilities and $\hat{x}$ clients. At this point the solution can be filtered, the following section describes the steps involved in filtering (Lin & Vitter, 1992; Solis-Oba, 2006).

3.2. Filtering

Filtering is the process of removing certain non-optimal solutions that can be easily identified while ensuring a better solution remains. The following filtering process was described by Lin and Vitter in 1992 and was the first approximation algorithm for the metric k-median problem. A cost for each client, $Cost_i = \sum_{i \in F} c_{ij} \hat{x}_{ij}$, can be defined as a fractional cost for the client. In this case, the filter only selects from facilities for which the c_{ij} is less than or equal to $(1 + \varepsilon)Cost_i$, where $\varepsilon > 0$. This requirement defines a client's neighbourhood which is composed of potential facilities that satisfy this filter. The filtered data can then be used as input into the greedy set cover algorithm, an approximation algorithm for the set cover problem. The k-median approximation algorithm using filtering is as follows:

Algorithm: Filtering

1. Take the integer linear program and generate a relaxation, solve the linear program as an optimization problem to get a solution for facilities and clients.
2. Calculate a sum of costs from all facilities, $Cost_i = \sum_{i \in F} c_{ij} \hat{x}_{ij}$, for each client i.
3. The greedy set cover algorithm is then used with ground set D of facilities from the LP solution, and a family of subsets formed from the neighbourhoods of each facility.
4. The greedy set cover algorithm will select a subset of facilities, choose those facilities and assign them as the facilities for each client, based on the closest facility to each client.
   (Stern, 2006; Lin & Vitter, 1992; Solis-Oba, 2006)

The centers selected by this algorithm may not be exactly k centers because the greedy set cover algorithm returns up to $\left(1 + \frac{1}{\varepsilon}\right)(\ln(n) + 1)k$ where the factor $\bar{s} = \left(1 + \frac{1}{\varepsilon}\right)$ is used to bring the number of centers as close to k as possible. The set cover problem returns by default $\bar{s}(\ln(n) + 1)$ centers (Lin & Vitter, 1992; Solis-Oba, 2006).

3.3 Constant Performance Ratio Approximation

The first algorithm to approximate the metric k-median with a constant performance ratio was devised by Charikar, Guha, Targos, and Shmoys in 1999. They modified the k-median integer LP problem by adding a positive demand d_i to the minimization.

(6)

$$Minimize \sum_{i,j \in N} d_i c_{ij} X_{ij}$$

This LP problem is known as a fractional k-median problem and it contains demands. Like the previous algorithm, after obtaining the optimal solution $\hat{x}$ and $\hat{y}$ we can calculate the fractional cost as $Cost_i = \sum_{i \in F} c_{ij} \hat{x}_{ij}$ per location i. After obtaining the fractional cost, we organize the costs for each client from lowest cost to highest cost. Finally, instead of separating the clients and facilities into separate groups we have one group N that represents all the locations (Charikar et al., 1999; Solis-Oba, 2006). Now the following algorithm can be run:

Algorithm: Consolidate

1. Solve the linear program and get the optimal solution $\hat{x}$ and $\hat{y}$
2. Reorganize the demands by creating a new variable d'_i and assign it the value of d_i. For each location i, if there is a location j that has a lower fractional cost with a demand d_j greater than zero and c_{ij} less than or equal to four times the fractional cost, then move the demand of i to j and make the demand of i zero.

3. Create a set of locations for which the demand is greater than zero, known as $N'_{>}$. Iterate through all locations i, find the location in $N'_{>}$ that is closest to the location i and assign $s(i)$ this value, if there are ties choose the lowest index location.
4. Create two variables to hold $\hat{x}$ and $\hat{y}$, named x' and y', respectively
   For every location that has a y'_i with a value bigger than zero and no demand
   a. Set the $y'_{s(i)}$ to the minimum between one and $y'_i + y'_{s(i)}$ followed by setting y'_i to zero
   b. For each location j set $x'_{j\,s(i)}$ to $x'_{j\,s(i)} + x'_{ji}$, followed by setting x'_{ji} to zero
5. Sort the locations i in $N'_{>}$ into decreasing order by values $d'_i c_{i\,s(i)}$
   Let $\bar{x}$ and $\bar{y}$ be the parameters that hold the optimal solution
   Set $\bar{y}_i$ to one for the first $2k - |N'_{>}|$ locations in the ordered list
   Set $\bar{y}_i$ to one-half for the rest of the $2(|N'_{>}| - k)$ locations in the ordered list
   Set $\bar{y}_i$ to zero for all locations that are in N but not in $N'_{>}$
   For each location i in N
   Set the value of the self-transportation quantity, $\bar{x}_{ii}$, to the value $\bar{y}_i$ and set the value of $\bar{x}_{i\,s(i)}$ to 1 minus $\bar{y}_i$
6. A graph can now be generated with the vertices being the locations in $N'_{>}$ (G = (V_H, E_H))
   For each vertex i with the value $\bar{y}_i$ equal to one-half, create an edge from $s(i)$ to i. The resulting graph will be acyclic with multiple trees, also known as a forest.
   Find a dominating set I for the graph, containing all the vertices with $\bar{y}_i$ equal to one. A dominating set such that $|I|$ is less than or equal to k
7. Finally select the set I to represent the centers and client i can be served by one of the centers, depending on the minimal cost for a center to a client c_{ij}.

First, this algorithm optimizes the LP problem, then reduces the resulting solution by identifying a set of solutions ($N'_{>}$) with positive demands of at most $2k$ locations. The solution is further distilled by sorting the solutions and removing candidate centers that have are significantly far away. This is done by separating assigning the values zero, one-half, or one to the candidate centers ($\bar{y}_i$), according to the ½-integral solution. From this, the integral solution is derived and the value of one-half, for the candidate centers ($\bar{y}_i$), can be rounded to one or zero by using a rounding technique. In this case, a more complex rounding technique can produce a more accurate approximation algorithm.

While analyzing this algorithm, it is apparent that the cost of the solution is at most eight times the cost of the optimum solution. There is a factor of two that arises in step four, the steps where centers are reassigned, in this step the ideal amount of transported product from facility to client is set to $x'_{j\,s(i)}$ which is the sum of $x'_{j\,s(i)}$ and x'_{ji}, in the worst case this sum will be twice $\hat{x}_{ij}$. Additionally, another factor of four arises from step two, the reallocation step, particularly the condition for step two checks if c_{ij} less than or equal to four times the fractional cost, in the worst case, this would allow the cost to remain four times the fractional value. As such, because of the factor of two and the factor of four the combined cost can be up to eight times the optimum cost (Charikar et al., 1999; Solis-Oba, 2006). The rounding method used in this algorithm is randomized rounding. A dependent rounding method developed using 3-level trees has a more accurate approximation ratio of $6\frac{2}{3}$. (Charikar et al., 1999). In 2012, Charikar and Li developed another dependent rounding technique, they were able to produce an approximation

algorithm for the k-median algorithm with a ratio of $3.25(1 + \delta)$ and a time complexity of $O(\frac{k^3 n^2}{\delta^2})$ (Czumak et al., 2012).

3.3. Primal-Dual Approximation Algorithm

In 1999, Jain and Vazirani developed a new approximation algorithm for the metric k-median problem by modelling it after the metric uncapacitated facility location problem. Although they used the LP descriptions of the k-median problem, they did not actually solve the problem using LP, rather they used a combinatorics approach. Linear programs are known to have two forms, the first being a maximization problem, known as the primal, and the second being a minimization problem, known as the dual. Each of these linear programs can be thought of as approaching an optimization problem from orthogonal directions to reach an optimized value. The dual metric k-median linear program, shown in equation (7), used by Jain and Vazirani is like the linear program we derived from the facility location problem in the introduction. The main difference is that there is no constraint on the number of facilities (k) because the langrange multiplier (z) can be used to tweak the number of facilities chosen, a larger z results in less facilities been chosen and vice versa, the optimal z value needs to be chosen for k facilities (Jain & Vazirani, 1999; Solis-Oba, 2006).

(7)

$$Minimize \sum_{j \in F} zY_j + \sum_{j \in F} \sum_{i \in CL} c_{ij} X_{ij}$$

$$constraints:$$

$$\forall i \in CL: \quad \sum_{j \in F} X_{ij} \geq 1$$

$$\forall j \in F \; \forall i \in CL: \quad X_{ij} \leq Y_j$$

$$\forall j \in F \; \forall i \in CL: \quad X_{ij}, Y_j \in \{0,1\}$$

The dual of this problem is solved using a matrix transpose of the linear program. It is defined as follows:

(8)

$$Maximize \sum_{i \in CL} \alpha_i - zk$$

$$constraints:$$

$$\forall j \in F \; \forall i \in CL: \;\; \alpha_i - \beta_{ij} \leq c_{ij}$$

$$\forall j \in F: \quad \sum_{i \in CL} \beta_{ij} \leq z$$

$$\forall j \in F \; \forall i \in CL: \quad \alpha_i, \beta_{ij} \geq 0$$

A connection of the dual to the primal can be made that says if a client is served by a facility, then the amount α_i paid by the client for the service cost is at least equal to the cost of transport c_{ij} from the facility to the client. The value β_{ij} is equivalent to the cost of building the selected facilities. If α_i is larger than the cost of transport c_{ij}, then the rest of the cost β_{ij} is the service paid by the client α_i minus the cost of transport c_{ij} and goes toward building the facility. In this case, β_{ij} is tight because it is bigger than zero according to the complementary slackness conditions. The formulation of the primal-dual algorithm is as follows:

Algorithm: Primal-Dual

1. Un-mark all clients. Initialize the dual parameters α_i and β_{ij} to zero.
2. While unmarked clients exist loop this and the next step.
3. The values of the dual variables α_i for unmarked vertices and β_{ij} for tight edges are increased linearly at the same time until:
    a. α_i is equal to the c_{ij} cost of transport from facility to client for some edge (i, j). In this case the edge is labelled as tight because β_{ij} is equal to zero.
    b. The sum of β_{ij} for all clients is equivalent to the Lagrange multiplier (z) or some f_j in the case of the metric facility location problem. Here the facility j is labelled as paid for.
4. Every unmarked client i with a tight edge to a paid for facility j is marked
5. Build a graph, $T = (CL \cup F, \; E')$, the new edges $E'$are the edges for which β_{ij} is larger than zero
6. Build a square graph T^2 by adding a new edge to any existing edges with value less than two between two vertices u and v.
7. Extract a subgraph from T^2 labelled H with the facilities that are paid for.
8. Find I, the maximal independent set of H
9. Select I as the facilities and assign the clients to the facilities based on minimum cost, if there are more facilities than one with minimum cost, select any of them.

(Jain & Vazirani, 1999; Solis-Oba, 2006)

The subgraph I has two results for clients, direct connection or indirect connection. If the value for connection β_{ij} is bigger than zero, then the client is directly connected to the facility, else it is indirectly connected. The performance ratio of this algorithm is three, this is because the indirectly connected clients may be connected through the triangle inequality to a specific facility and the value of the cost between the client and the facility must be less that $3\alpha_i$, according to the inequality. The time complexity of this algorithm is derived from the complexity

of steps one to three and is $O(mlog(n_f))$, where m is the number of edges in the graph and n_f is the number of facilities. This analysis is based on the case in step three where a facility j is paid for, so all the clients i with tight edges to j are marked and their α_i and β_{ij} stops increasing, we need to maintain the number of tight edges for j (b_j), the time the tight edges change (t_j), and the total contribution from clients to facility j (p_j). Updating these values since every client is marked at least once is $O(\sum_{i \in CL} degree(i)\ log(n_f))$. In the worst case, one less than all the clients are marked in this case, as a result, updating these values takes time of $O(mlog(n_f))$.

The k-median problem has an additional algorithm that is used to identify the ideal Lagrange multiplier (z). This algorithm produces a convex combination of two solutions, one greater and one less than k to isolate a fractional solution with exactly k-medians.

Algorithm: Convex Combination

1. Get the minimum and maximum cost edges labelled as c_{min} and c_{max}
2. Run the previously defined Primal-dual algorithm and binary search on interval $[0, nc_{max}]$ to find two values, z_1 and z_2, such that the former minus the latter is less than or equal to $c_{min}/(4n_f^2)$ for which the primal-dual algorithm can find solutions that meet the following conditions:
    a. The first solution z_1 produces a solution s $(x^s, y^s, \alpha^s, \beta^s)$ with k_1 centers less than k centers. Let A be the set of centers chosen by this solution.
    b. The second solution z_2 produces a solution l $(x^l, y^l, \alpha^l, \beta^l)$ with k_2 centers more than k centers. Let B be the set of centers chosen by this solution.
3. Define two variables a and b that are equal to $(k_2 - k)/(k_2 - k_1)$ and $(k - k_1)/(k_2 - k_1)$, respectively. The biconvex solutions can be resolved as follows to generate a fractional solution for exactly k centers, $(\hat{x}, \hat{y}) = a(x^s, y^s) + b(x^l, y^l)$.
4. Create an empty set B'
   For each facility j in A
   Remove the facility j' in B that has the smallest cost $c_{j\ j'}$ and add it to set B'
5. Now we will select the set of centers I of k centers from the union of the sets A and B.
    a. Select all the centers in A that have a probability a and all centers in B' that have probability b which is $1 - a$
    b. For the centers in B select $k - k_1$ at random
6. Use the I centers as facilities and connect the clients to the facilities with minimum cost.

The fractional solution that is retrieved in step three has a maximum approximation ratio of $3 + \frac{1}{n_f}$, this value arises from the factor of three in the primal-dual algorithm. An added factor of $2 - \frac{1}{n_f}$ occurs based on the solution for the convex combination algorithm. The overall performance ratio of this algorithm is $(3 + \frac{1}{n_f})(2 - \frac{1}{n_f})$ which is less than or equal to six. The time complexity of the previous algorithms is much higher than the time complexity of this algorithm because the other algorithms optimize the linear program and use other approximation algorithms in their solution. For this algorithm the time complexity is of the order $O(mlogm(L + \log(n))$, where n and m are the vertices and edges respectively and L is the number of bits needed to represent a connection cost (Jain & Vazirani, 1999; Solis-Oba, 2006). Jain et al., improved this

algorithm by using dependent rounding generating an approximation ratio of four for a time complexity of $O(n^3)$ (Jain et al., 2003.

3.4 Pseudo-approximation algorithm

Li and Svensson used the principle of bi-point solutions, the bi-point solution is essentially two possible solutions that surround k, for example $k+1$ and $k-1$. The idea is to remove facilities one at a time from some bi-point solution to reach an optimal value for k. The algorithm is referred to as the star algorithm because it uses star structures to optimize lowest cost from one facility to another.

The algorithm is defined as an r-pseudo-approximation algorithm that opens $k+c$ facilities, this algorithm can be turned into an $r+\varepsilon$ approximation algorithm that opens k facilities and runs in time $n^{O(c/\varepsilon)}$. There exists a $(1+\sqrt{3}+\varepsilon)$-pseudo-approximation algorithm that opens $O(1/\varepsilon)$ centers. Need to get $O(1/\varepsilon)$ close to k. The algorithm looks for sparse instances, and then removes a sparse instance such that the cost incurred to the solution is minimized.

The Jain and Vazirani primal-dual bi-point solution is obtained using the first two steps of the convex combination algorithm, this adds a factor of two to the approximation ratio based on the approximation ratio proposed by Charikar and Guha. The bi-point collection forms a set of stars as follows:

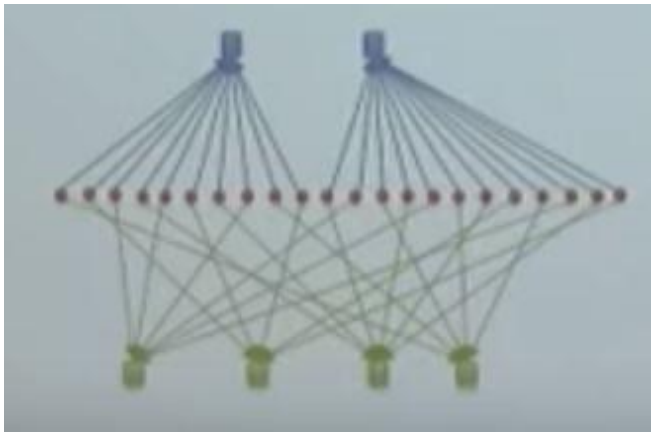

Figure 1. *Bi-point solutions from the primal-dual algorithm organized in such a way as to emphasize the connection structure*

Each facility in the leaf set is then connected to a facility in the root set with the lowest cost, the rest of their connections are removed.

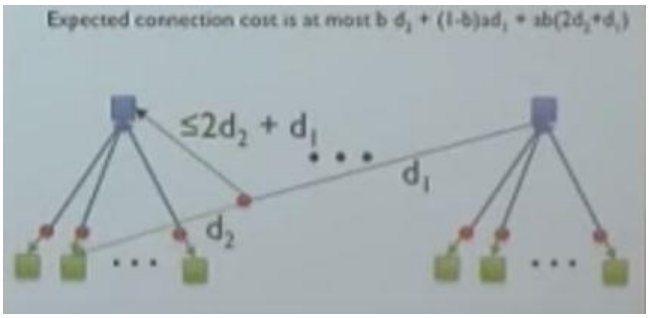

Figure 2. *Bi-point solutions with ideal connections selected*

This step adds an approximation factor of $\frac{1+\sqrt{3}}{2}$, making the total approximation ratio equal to $\frac{1+\sqrt{3}}{2} * 2$ which is equal to $1 + \sqrt{3}$. Recall that this algorithm is an approximation scheme and as such there is an added value of epsilon that will vary depending on the precision required. The time complexity of this algorithm is $O(n^{O(1/\varepsilon^2)})$, as a result the run-time is exponential with lower values of epsilon. This algorithm can produce an approximation ratio that is closest to the hardness of $1 + \frac{2}{e}$ for the metric k-median problem (Li & Svensson, 2012).

4. Discussion and Conclusion

In this section, the time and approximation ratio tradeoff will be discussed. Furthermore, proposed improvements and open questions for the k-median solutions will be reviewed. To begin, we will analyze the metric k-median problem approximation algorithms for two ideal scenarios. The first scenario is the minimization of the running time of the algorithm, if the lowest running time k-median approximation algorithm is desired then the primal-dual algorithm by Jain and Vazirani in 1999 with a run time of $O(mlogm(L + \log(n))$ can be selected, where n and m are the vertices and edges respectively and L is the number of bits needed to represent a connection cost. However, the approximation ratio of six for this solution is far from ideal. When seeking the best approximation ratio, one could use the pseudo-approximation algorithm with an approximation ratio of $1 + \sqrt{3} + \varepsilon$. An epsilon of less than 0.2679 or $3 - (1 + \sqrt{3})$ is needed to get a better approximation than the long-standing approximation ratio of $3 + \frac{2}{p}$ that was established by the local search p-swap algorithm. The time complexity for this algorithm with such an epsilon value will be $O(n^{O\left(\frac{1}{0.27679^2}\right)})$ which is approximately $O(n^{14})$. Of course, the closer epsilon is to zero the larger the run time will be, leading to a tradeoff problem. If an intermediate solution between low run-time and increased accuracy of approximation is required, either the local search p-swap algorithm or the 4-approximation ratio primal-dual algorithm by Jain et al., with time complexity $O(n^3)$ can be used.

The location search algorithm and LP algorithms can be combined such that the location search occurs after LP is complete, so that after the linear program has isolated a specific subset of potential facilities the location search approximation ratio of $3 + \frac{2}{p}$ can look for optimal solutions within this selected subset. For the primal-dual problem, when a convex combination of two solutions was used, the algorithm approximated the value of k facilities with a ratio of 6.

Creating a convex-combination algorithm with more than two primal-dual solutions, may help to further narrow the approximation ratio

For the metric k-median problem, a hardness limit of $1 + \frac{2}{e}$ exists, how did such a hardness limit come to exist? Is there a way to approach this set limit through some procedure? The hardness limit can be written based on the expansion of the Euler number as follows:

(9)

$$1 + 2 \lim_{n \to \infty} (1 + \frac{1}{n})^{-n}$$

In terms of approximation ratio, it seems that the Euler expansion gives us some clues as to the direction of investigation. First and foremost, there is an iterative component as defined by the limit, the iterative component gets closer and closer to one over Euler's number as n approaches infinity. There are multiple applications of Euler's number that stand out immediately particularly its use in statistical distributions, it is often to the negative power. Additionally, the usage of Euler's number in quantum mechanics for wave functions is another interesting observation. It would be worth further investigating the nature of this hardness limit to better understand the value as it currently stands. Technically, the hardness is limited to values that are less than $1 + \frac{2}{e}$.

Another potential idea for future research is the k-nearest neighbor algorithm, the algorithm identifies the k-nearest neighbours to a node in a graph. The total number of nodes n could be divided by the number k, to give an average number, f, of nodes per median. Then a set of all the f-nearest neighbours can be calculated and a set cover algorithm can be used to reduce this to the minimum covering set of the graph.

Finally, many of these problems have very long run times and this is mostly due to the iterative or repetitive nature of finding a linear programming solution. As such, a computer that has an advantage in its' ability to perform parallel computations quickly, and to represent an exponential number of states may prove beneficial to many of these problems. A couple of these solutions have subroutines that can be encoded-based on amplitude and run on quantum computer. The technology is in its' preliminary phases, however, things that were near impossible before such as modelling protein-folding and chemistry interactions are now simulated on quantum computers. The benefit of superposition, and entanglement in a quantum computer cannot be understated.

Overall, the k-median problem is a difficult problem and will require a strong effort with multiple different approaches to optimize, the number of solutions over the past three decades demonstrate the need for a well approximated solution.

References

Andrews, M., & Zhang, L. (n.d.). The access network design problem. *Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No.98CB36280)*. doi:10.1109/sfcs.1998.743427

Arya, V., Garg, N., Khandekar, R., Munagala, K., & Pandit, V. (2001). Local search heuristic for k-median and facility location problems. *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing - STOC 01*. doi:10.1145/380752.380755

Bartal, Y. (n.d.). Probabilistic approximation of metric spaces and its algorithmic applications. *Proceedings of 37th Conference on Foundations of Computer Science*. doi:10.1109/sfcs.1996.548477

Bradley, P. S., Fayyad, U. M., & Mangasarian, O. L. (1998). Mathematical Programming for Data Mining: Formulations and Challenges. *INFORMS Journal on Computing,11*(3), 217-238. doi:10.1287/ijoc.11.3.217

Charikar, M., Chekuri, C., Goel, A., & Guha, S. (1998). Rounding via trees. *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing - STOC 98*. doi:10.1145/276698.276719

Charikar, M., & Guha, S. (n.d.). Improved combinatorial algorithms for the facility location and k-median problems. *40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039)*. doi:10.1109/sffcs.1999.814609

Charikar, M., Guha, S., Tardos, E., & Shmoys, D. (2002). A constant-factor approximation algorithm from the k-Median problem. *Journal of Computer and System Sciences, 65(1),* 129-149. doi: 10.1006/jcss.2002.1882

Czumaj, A., Mehlhorn, K., Pitts, A., & Watterhofter, R. (2012). *Automata, Languages, and Programming.* Retrieved from https://link.springer.com/content/pdf/10.1007%2F978-3-642-31594-7.pdf?fbclid=IwAR3Oe85RRM5w3jdhH4vMoF3KIvSSqEe2XmJcb9JMeB8hjBEDx_fhmmx2dHs.

Jain, K., & Vazirani, V. (1999). Primal-dual approximation algorithms for metric facility location and k-median problems. *Proceedings of the 40th Annual Symposium on Foundations of Computer Science,* 2-13. doi: 10.1109/SFFCS.1999.814571

Jain, K., Mahdian, M., Markakis, E., Saberi, A., & Vazirani, V. V. (2003). Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *Journal of the ACM,50*(6), 795-824. doi:10.1145/950620.950621

Kuehn, A. A., & Hamburger, M. J. (1976). A Heuristic Program for Locating Warehouses. *Lecture Notes in Economics and Mathematical Systems Mathematical Models in Marketing,*406-407. doi:10.1007/978-3-642-51565-1_123

Li, S., & Svensson, O. (2012). Approximating k-Median via pseduo-approximation. *Proceedings of the forty-fifth annual ACM symposium on Theory of computing,* 901 – 910. doi: 10.1137/130938645

Lin, J., & Vitter, J. (1992). Approximation algorithms for geometric median problems. *Information Processing Letters, 44(5),* 245-249, doi: 10.1016/0020-0190(92)90208-D

Mulvey, J. M., & Crowder, H. P. (1979). Cluster Analysis: An Application of Lagrangian Relaxation. *Management Science,25*(4), 329-340. doi:10.1287/mnsc.25.4.329

Solis-Oba, R. (2006). Approximation algorithms for the k-Median problem. Lecture Notes in Computer Science Efficient Approximation and Online Algorithms, 292-320. doi: 10.1007/11671541_10

Stern, T. (2006). *Set Cover Problem (Chapter 2.1, 12).* Retrieved from https://math.mit.edu/~goemans/18434S06/setcover-tamara.pdf