Optimization of P Median Problem in Python Using PuLP Package

Anand Jayakumar Arumugham
Sri Ramakrishna Institute of Technology

Aravinth Kumar Annamalai
PSG College of Technology

IJCTA, 10(02), 2017, pp. 437-442

Optimization of P Median Problem in Python Using PuLP Package

Anand Jayakumar A* Krishnaraj C and Aravith Kumar A*****

Abstract : Location planning involves specifying the physical position of facilities that provide demanded services. There are a variety of different models to solve this problem The *p*-median problem is a specific type of a discrete location model. In this model, we wish to place *p* facilities to minimize the (demand-weighted) average distance between a demand node and the location in which a facility was placed. This serves as an approximation to total delivery cost. In this model, there are no capacity constraints at the facilities. In this paper a real time case study is solved using PuLP package in Python.

Keywords : P-median problem, Case study, Python.

1. INTRODUCTION

The *p*-median problem is one of a larger class of problems known as minisum location-allocation problems. These problems find medians among existing points, which is not the same as finding centers among points, a characteristic of minimax location-allocation problems (the *p*-center problem is an example, where the goal is to minimize the maximum distance between points and center(s)). Minisum problems originated in the 17th century when Fermat posed the following question: Given a triangle (three points in a plane), find a median point in the plane such that the sum of the distances from each of the points to the median point is minimized. In the early 20th century, Alfred Weber presented the same problem with the addition of weights on each of the three points to simulate customer demand. Finding the median point corresponded to finding the best location for a facility to satisfy the demands at the points. This problem is usually acknowledged as the first location-allocation problem. It was later generalized to find the median of $n \geq 3$ points in a plane, and to the multifacility Weber problem, which generalizes to the case of $p > 1$ medians among a number of points in the plane.

Facility-location problems have several applications in telecommunications, industrial transportation and distribution, etc. One of the most well-known facilitylocation problems is the p-median problem. This problem consists of locating p facilities in a given space (*e.g*. Euclidean space) which satisfy *n* demand points in such a way that the total sum of distances between each demand point and its nearest facility is minimized. In the noncapacitated p-median problem, one considers that each facility candidate to median can satisfy an unlimited number of demand points. By contrast, in the capacitated *p*-median problem each candidate facility has a fixed capacity, *i.e*. a maximum number of demand points that it can satisfy. The *p*-median problem is NP-hard. Therefore, even heuristic methods specialized in solving this problem require a considerable computational effort.

In this paper we solve a P Median problem using PuLP package in Python.

* Dept of Mechanical Engineering SVS College of Engineering Coimbatore, India. *jay4upeople@gmail.com*

** Dept of Mechanical Engineering Karpagam College of Engineering Coimbatore, India. *krishna.kce@gmail.com*

*** Dept of Mechanical Engineering United Institute of Technology Coimbatore, India. *aravinthkumar004@gmail.com*

2. LITERATURE REVIEW

Pasquale Avella et al (1), have developed a new heuristic for large-scale PMP instances, based on Lagrangean relaxation. Computational results show that the proposed heuristic is able to compute good quality lower and upper bounds for instances up to 90,000 clients and potential facilities. Landa-Torres et al (2), have paper addresses the application of two different grouping-based algorithms to the so-called capacitated P-median problem (CPMP). Jack Brimberg and ZviDrezner (3), have presented a new local search for solving the continuous p-median problem in the plane. ZviDrezner et al (4), have proposed an effective heuristics for the solution of the planar p-median problem. FatemehSayyady et al (5), have also proposed a Lagrangian heuristic algorithm for solving larger instances of this problem. G.J. Lim et al (6), have discussed new solution techniques for the p-median problem, with the goal being to improve the solutiontime and quality of current techniques. Xiang Li et al (7), focused on the strategies of generating the initial population of a genetic algorithm and examined the impact of such strategies on the overall GA performance in terms of solution quality and computationaltime. Gino J. Lim and Likang Ma (8), have introduced a GPU-based parallel vertex substitution (pVS) algorithm for the p-median problem using the CUDA architecture by NVIDIA. Juan A. Dı´az and Elena Ferna´ndez (9), have proposed heuristic algorithms for the capacitated p-median problem. Laura E. Jackson et al (10), gave an improved solution through reformulating the problem as a special case of the constrained shortest path problem.

3. THE MODELING FRAMEWORK

In this paper a mathematical formulation by G. Srinivasan (14), is considered. Our present work for P Median Location problem is largely inspired from this work.

Where d_{ij} = distance between the points.

X_{ij} = whether the distance is selected.

p = number of median points.

$$\text{Minimize } \sum_{i=1}^{n}\sum_{j=1}^{n} d_{ij}X_{ij}$$

Subject to

$$\sum_{j=1}^{n} X_{ij} = p$$

$$\sum_{j=1}^{n} X_{ij} = 1$$

$$X_{ij} \leq X_{ij}$$

$$X_{ij} = 0, 1_{j}$$

4. NUMERICAL PROBLEM

Table 1
Distance Between the Warehouses

	Kurudampalayam	*Ashokapuram*	*Peelamedu*	*Vellalore*	*KovaiPudur*	*Vadavalli*
Kurudampalayam	0	5.6	17.4	21.8	19.4	11.4
Ashokapuram	5.6	0	11.6	19.6	21.7	13
Peelamedu	17.4	11.6	0	8	18.3	16.8
Vellalore	21.8	19.6	8	0	16.9	20.1
KovaiPudur	19.4	21.7	18.3	16.9	0	13.6
Vadavalli	11.4	13	16.8	20.1	13.6	0

The study being reported here was carried out in a company situated in Coimbatore city, Tamil Nadu State, India. As the management of this company prefers to maintain anonymity, this company is referred to in this paper as XYZ. XYZ has six warehouses in Coimbatore. The management wants to know from which depots they have to supply to other depots such that the distance is minimized.The distance between the warehouses is given in Table 1. Three scenarios were analyzed here. Having the central depots as 1, 2 and the results were presented before the management.

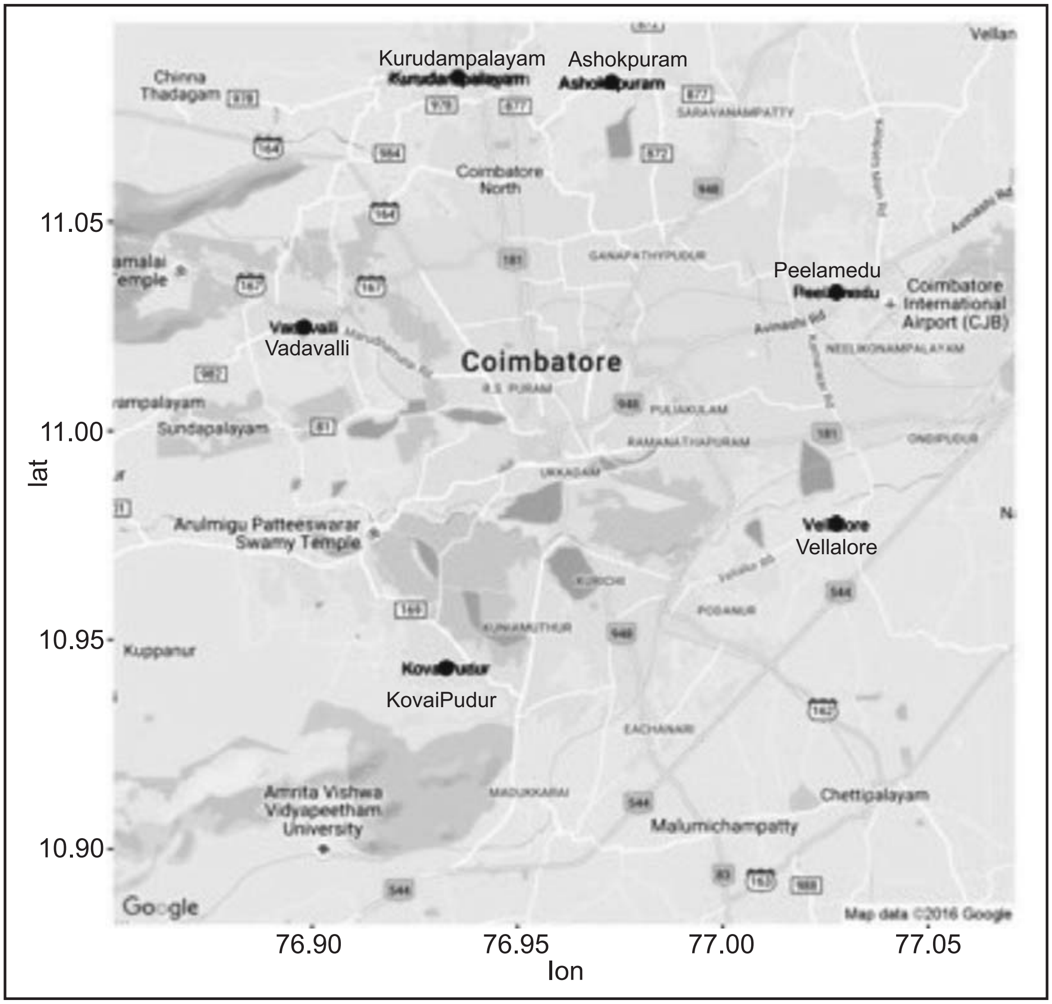

Figure 1. Location of the Depotst

5. PYTHON PROGRAM

 # import library

from pulp import *

 # variable assignment

location = [‘Kurudampalayam’,’Ashokapuram’,’Peelamedu’,’Vellalore’,’KovaiPudur’,’Vadavalli’]

D = dict(zip(location,[dict(zip(location, [0, 5.6, 17.4, 21.8, 19.4, 11.4])),
dict(zip(location, [5.6, 0, 11.6, 19.6, 21.7, 13])),
dict(zip(location, [17.4, 11.6, 0, 8, 18.3, 16.8])),
dict(zip(location, [21.8, 19.6, 8, 0, 16.9, 20.1])),
dict(zip(location, [19.4, 21.7, 18.3, 16.9, 0, 13.6])),
dict(zip(location, [11.4, 13, 16.8, 20.1, 13.6, 0]))]))

p = 2

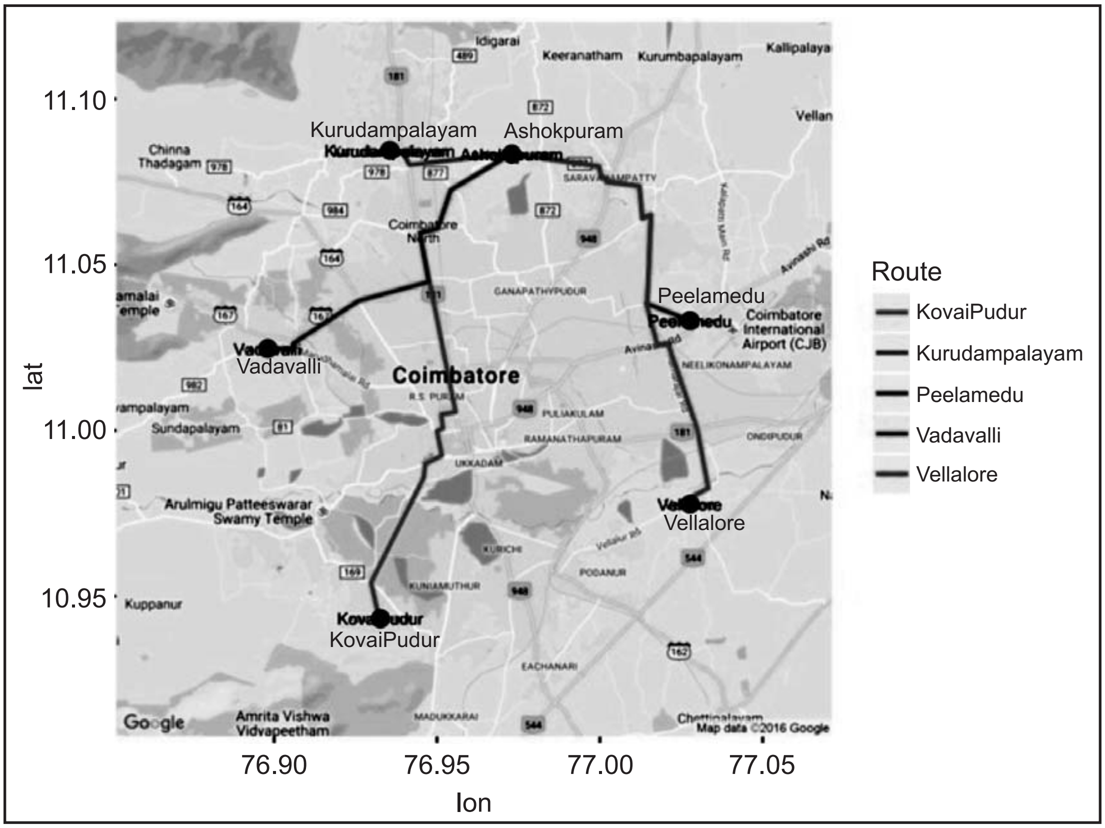

Figure 2: Scenario 1

 # decision variables

X = LpVariable.dicts('X_%s_%s', (location,location),

cat = 'Binary',

low Bound = 0,

up Bound = 1)

 # create the LP object, set up as a MINIMIZATION problem

prob = LpProblem('P Median', Lp Minimize)

prob + = sum(sum(D[*i*][*j*] * X[*i*][*j*] for *j* in location) for *i* in location)

 # setup constraints prob + = sum(X[*i*][*i*] for *i* in location) == *p*

for *i* in location : prob + = sum(X[*i*][*j*] for *j* in location) == 1

for *i* in location:

for j in location: prob + = X[*i*][*j*] <= X[*j*][*j*]

 # save the model to a lp file

prob.writeLP("p-median.lp")

 # view the model

print(prob)

 # solve the model

prob.solve()

print("Status:",LpStatus[prob.status])

print("Objective: ",value(prob.objective))

for v in prob.variables():

print (v.name , "=", v.varValue)

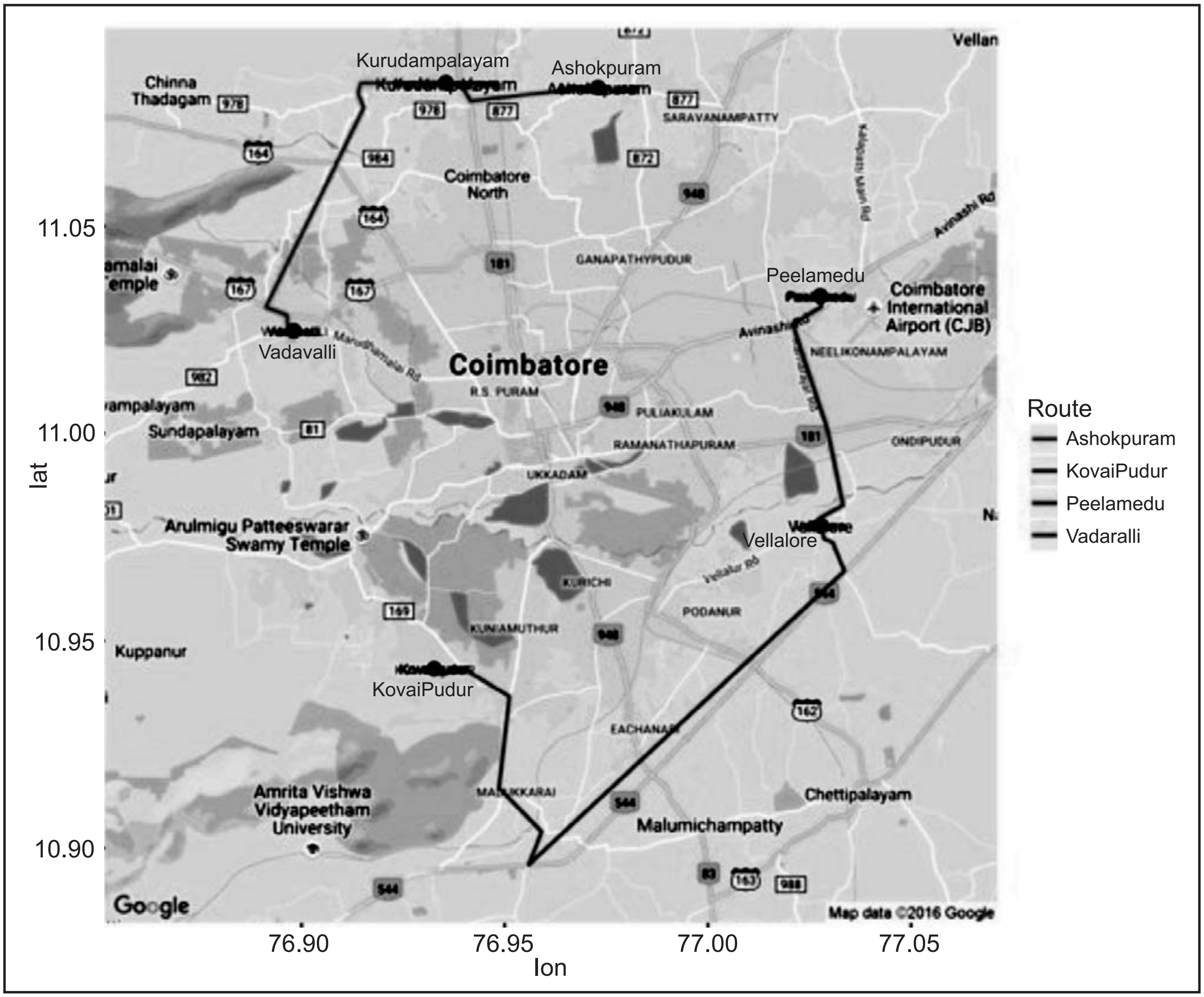

Figure 3: Scenario 2

6. COMPUTATIONAL EFFICIENCY

An intel CORE i5 processor 2nd Generation with 4GB RAM was used to process the model.The operating system used was Windows 7. Python 3.5.2 :: Anaconda 4.2.0 was used. PuLP package 1.6.1 was used. The default solver was CBC.

The problem was solved in less than 1 second.

7. RESULT AND DISCUSSION

For scenario 1 the management wants to check with one depot as the central depot. 71.5 kms was the distance travelled with Ashokapuram as the central depot.

For scenario 2 the management wants to check with two depots as the central depot. 41.9 kms was the distance travelled with Kurudampalayam and Vellalore as the central depots. Kurudampalayam would supply Ashokapuram and Vadavalli and Vellalore would supply Peelamedu, and KovaiPudur.

8. CONCLUSION

Thus we have successfully analyzed all the scenarios using Python PuLP package.. The best solution was also found

9. REFERENCES

1. Pasquale Avella, MaurizioBoccia, SaverioSalerno, IgorVasilyev, "An aggregationheuristicforlargescale p-median problem", Computers & Operations Research vol39, pp 1625–1632, 2012.
2. I. Landa-Torres, J.DelSer, S.Salcedo-Sanz, S.Gil-Lopez, J.A.Portilla-Figueras, O.Alonso-Garrido, "A comparative study of two hybrid grouping evolutionary techniques for the capacitated P-median problem", Computers & Operations Research vol39, pp 2214–2222, 2012.
3. Jack Brimberg, ZviDrezner, "A new heuristic for solving the p-median problem in the plane", Computers & Operations Research, Vol 40, pp 427–437, 2013.
4. ZviDrezner, JackBrimberg, NenadMladenović, SaidSalhi, "Newheuristicalgorithmsforsolvingtheplanar p-median problem", Computers &OperationsResearch, Vol 62, pp 296–304, 2015.
5. FatemehSayyady, GolbargK.Tutunchi, YahyaFathi, "p-Median and p-dispersion problems:A bi-criteria analysis", Computers &OperationsResearch, Vol 61, pp 46–55, 2015.
6. G.J. Lim, J. Reese, A. Holder, "Fast and robust techniques for the euclidean p-median problem with uniform weights", Computers & Industrial Engineering, Vol 57, pp 896–905, 2009.
7. Xiang Li, Ningchuan Xiao, Christophe Claramunt, Hui Lin, "Initialization strategies to enhancing the performance of genetic algorithms for the p-median problem", Computers & Industrial Engineering, Vol 61, pp 1024–1034, 2011.
8, Gino J. Lim, Likang Ma, "GPU-based parallel vertex substitution algorithm for the p-median problem", Computers & Industrial Engineering, Vol 64, pp 381–388, 2013.
9. Juan A. Dı´az, Elena Ferna´ndez, "Hybrid scatter search and path relinking for the capacitated p-median problem", European Journal of Operational Research, Vol 169, pp 570–585, 2006.
10. Laura E. Jackson, George N. Rouskas, Matthias F.M. Stallmann, "The directional p-median problem: Definition, complexity, and algorithms", European Journal of Operational Research, Vol 179, pp 1097–1108, 2007.
11. Krishnaraj, C., A. Anand Jayakumar, S. Deepa Shri, "Solving Supply Chain Network Optimization Models Using LINGO", International Journal of Applied Engineering Research, Vol 10, No 19, pp14715-14718, 2015
12. Anand Jayakumar, A., C. Krishnaraj, "Solving Supply Chain Network Gravity Location Model Using LINGO", International Journal of Innovative Science Engineering and Technology", Vol 2, No 4, pp 32-35, 2015.
13. Anand Jayakumar A, Krishnaraj C, Aravinth Kumar A, "LINGO Based Supply Chain Network Design", Journal of Applied Sciences Research, Vol 11, No 22, pp 19-23, Nov 2015.
14. G Srinivasan, "Quantitative Models in Operations Management", PHI.