Visualisation Geographic Information: Finding Representative Points of Interest

João Valença
University of Coimbra
Coimbra, Portugal
valenca@student.dei.uc.pt

Luís Paquete
University of Coimbra
Coimbra, Portugal
paquete@dei.uc.pt

Pedro Reino
Smartgeo Solutions
Lisboa, Portugal
pedro.reino@smartgeo.pt

Carlos Caçador
Smartgeo Solutions
Lisboa, Portugal
carlos.cacador@smartgeo.pt

Abstract

Currently, there is a need to process a large amount of geographic data before it can be managed and analysed. This project aims to develop a GIS application operating through a Web platform in order to allow for a low cost and simplified integration, management and manipulation of georeferenced information. In particular, the goal is to develop a way to efficiently extract a subset of a collection of geographic points of interest whilst keeping the representativeness as close as possible to the whole set. This problem of finding such representation is recast as a discrete optimisation problem. The approaches covered in this work include exact algorithms for finding minimum coverage subsets, as well as heuristic approaches for finding good approximations in real time.

Keywords: Geographic Clustering, k-Center, Coverage, Branch-and-Bound, Delaunay Triangulations

1 Objective

One obstacle when representing large amounts of geographic data is that the sheer volume of points to display can be overwhelming for a human, as well as computationally intensive to render for a machine. As such, there is a need to develop and implement a viable way to reliably calculate and display a subset of geographic points, whilst keeping a degree of representativeness of the larger set, so that as little information as possible is absent when the representative subset is shown.

The purpose of this project is to develop a real-time algorithmic framework that can analyse geographic data provided by a geographic information system infrastructure. More precisely, the developed algorithm has to be able to aggregate and select geographic points according to a given a set of criteria in real time. The chosen subset should keep a measure of representativeness of the larger set.

Figure 1 shows a representative subset of the set of cities towns and villages in Sweden. The points were taken from a TSP Dataset from the University of Waterloo [3].

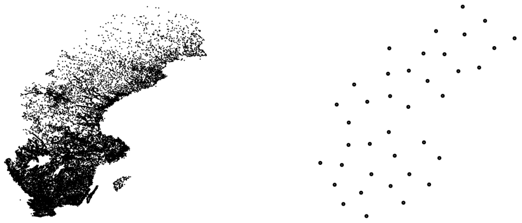

Figure 1: Representative Subset of Sweden's TSP Dataset [3]

2 Problem Definition

Representativeness consists of finding a subset of points in a larger set that meets some representation quality. The subset chosen should be able to keep some properties of the original set, such as density, or general distribution.

In this work, representativeness is understood as finding the subset that minimises the maximum distance between the points not chosen, and their closest counterparts within the chosen set. This notion of representativeness is known in discrete optimisation as coverage. The problem can be approached by finding the subset of cardinality k that minimises the value of coverage, known as the *k-center* problem [2]. Another approach is to establish an arbitrary minimum distance and minimise the number of points selected from the original set. This approach can be cast as a set cover problem.

In order to address these problems, a few algorithms were developed and implemented. Optimal approaches to the problems include integer linear programming, as well as incremental branch-and-bound algorithms. However, the inherent overhead in these algorithms makes them unsuitable for use in practical applications. For achieving an acceptable approximation to the optimal solution within a reasonable time frame of what is expected in a real-time web application, heuristic methods need to be used.

3 Architecture

The application displays a rectangular window, showing a cut of geographical region containing a set of points. The algorithm chosen needs to be able to choose a representative set of points within the cut quickly, as well as be able to recalculate a new set points for a new cut, resulting from panning or zooming the display window over the region.

The algorithms tested include exact algorithms for finding the minimum coverage subset for a given cardinality. These are comprised of two branch-and-bound approaches: a naïve incremental approach and a geometric incremental approach that makes use of the properties of Delaunay triangulations in order to speed up point location queries via the Greedy Routing algorithm [1]. An exact integer linear programming approach is benchmarked as well. This project also includes a few heuristic approaches and approximation algorithm techniques to solve the representation problem more efficiently.

The algorithm serves as the middleware responsible for filtering the response of a GIS server to a Web Feature Service, or WFS request. Finally, a web application receives the response filtered by the algorithm and interacts with a human user.

The candidate algorithms will be tested and benchmarked using data from the Open Street Map project. The project features large quantities of open source geographic data, as well as a versatile API for fetching data.

4 Results

The first approach solves the representativeness problem by finding the minimum the coverage subset with a fixed cardinality. The solutions include Branch-and-Bound algorithms, as well as Integer Linear Programming. The performance of these algorithms is too slow even for a small number of points, and takes minutes to solve very small instances of the problem, deeming these algorithms unusable in the context of a web application.

The second approach, i.e. minimising the number of points chosen given a fixed minimum distance, was solved using a heuristic approach to the set cover problem. This algorithm yielded the much faster times, with an acceptable quality for the resulting sets. Figures 2 and 3 plot these results.

These results were obtained by running the algorithm on a square region with uniformly distributed points. A minimum distance was fixed (as a percentage of the region size) for the different runs. For each run, the number of points was varied to plot the CPU time as a function of N. Figure 2 shows the CPU time the algorithm in seconds. Figure 3 shows the number of points selected. The algorithm used to produce these results was a set coverage approximation algorithm, which guarantees a result with at worst $\log_2 N$ more points than the optimal solution.

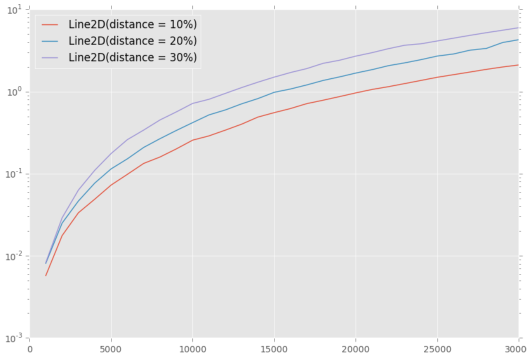

Figure 2: CPU time for various values of N

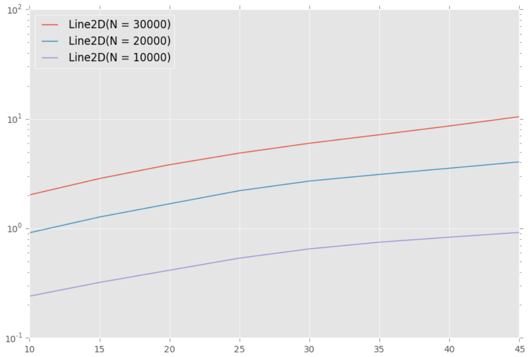

Figure 3: Number of points selected for different values of N

Acknowledgements

This work is being developed within the work plan of a research grant for undergraduates, in the context of the R&D project QREN unit in co- promotion no.34164 - Smartgeo Portal ”Geographic Information Management System”. The project is co-financed by QREN and Innovation Agency within the scope of the technological development and research incentive system, by the Operational Program for Competitiveness and Internationalizations (Compete).

References

[1] P. Bose and P. Morin. Online routing in triangulations. *Algorithms and Computation, LNCS*, 1741:113–122, 1999.

[2] N. Megiddo and A. Tamir. New results on the complexity of p-centre problems. *SIAM Journal on Computing*, 12(4):751–758, 1983.

[3] M. D. of the University of Waterloo. Travelling salesman problem datasets - sweden. URL `http://www.math.uwaterloo.ca/tsp/world/sw24978.`