

1 Introduction

The map overlay problem has been a classic topic with several applications in fields such as Cartography, Geographic Information Systems and Computer Graphics. The input of a map overlay operation are two independent layers of polygons. The results is another polygon layer which carries new information not possible to infer from the original layers. For example, one initial layer could represent the population at each county in the state of California. Another layer could show the areas prone to natural hazards like floods or forest fires. The overlay of the two layers could bring important information about the number of affected people and the best location for shelters or evacuation roads.

Nowadays, the rise of big (spatial) data have allowed the collection and access to huge amount of data. For instance, USGS provides access to large datasets about hidrography features and landcover for the entire US which ranges several Gigabytes of data. In addition, several sensitive applications, such as emergency response or environmental modelling, demands quick responses from the processing of overlay operations. The increasing volume of data to be processed and the time-sensitive needs for the results requires new techniques in order to improve the performance of map overlay operations.

Most of the techniques to solve this problem are sequential and they has been implemented in desktop packages. Even though the use of spatial-aware data structure and indexing, the performance and confiability of such solutions are still poor. Depending the extend and complexities of the input layers, the results can take several hours or, frequently, exceeds the resources of the system.

Lately, multi-core architecture and cloud-computing platforms have emerged as an interesting alternative. Indeed, some approaches have explored the use of GPUs or distributed frameworks to explore the problem of scalable parallel overlay computation. However, none of them seems to take advantage of intermediate data structures which allow multiples queries once they have been built.

The doubly connected edge list (DCEL), also know as half-edge data structure, is a spatial data structure which collect the topological information of planar graphs in the plane. Due to the input layers of polygons can be represented as a planar graphs (a planar subdivision of the space), the DCEL is a suitable format to use during the map overlay operations. The main reason to integrate a DCEL in the solution is the chance to run multiples types of overlay operations over it without the need to re-process the input data.

This paper addresses the implementation of a parallel DCEL construction to support scalable big spatial data overlay operations.