
Classifying Cartographic Projections Based on Dynamic Analysis of Program Code

Florian Ledermann ^{a,*}

^a *FB Kartographie, TU Wien, Florian Ledermann - florian.ledermann@tuwien.ac.at*

* Corresponding author

Abstract:

Analyzing a given map to identify its projection and other geometrical properties has long been an important aspect of cartographic analysis. If explicit information about the projection used in a particular map is not available, the properties of the cartographic transformation can sometimes be reconstructed from the map image. However, such a process of projection analysis requires significant manual labor and oversight.

For digital maps, we usually expect the projection from geographic space to map space to have been calculated by a computer program. Such a program can be expected to contain the implementation of the mathematical rules of the projection and subsequent coordinate transformations such as translation and scaling. The program code, therefore, contains information that would allow an analyst to reliably identify map projections and other geometrical transformations applied to the input data.

In the case of interactive online maps, the code generating the map *is* in fact delivered to the map user and could be used for cartographic analysis. The core idea of our novel method proposed for map analysis is to apply reverse engineering techniques on the code implementing the cartographic transformations in order to retrieve the properties of the applied map projection. However, automatic reasoning about computer code by way of *static analysis* (analyzing the source code without running it) is provably limited – for example, the code delivered to the map user may contain a whole library of different map projections, of which only a specific one may be actually used at runtime. Instead, we propose a *dynamic analysis* approach to observe and monitor the operations performed by the code as the program runs, and to retrieve the mathematical operations that have been used to calculate the coordinates of every graphical element on the map.

The presented method produces, for every graphical element of the map, a *transformation graph* consisting of low-level mathematical operations. Cartographic projections can be identified as distinctive *patterns* in the transformation graph, and can be distinguished in a fully automatic way by matching a set of predefined patterns against a particular graph.

Projections vary widely in their arithmetic structure, and therefore by the structure of the corresponding transformation graphs extracted from program code. Some projections can be computed directly using continuous equations involving trigonometric functions. Other projections involve solving nonlinear equations, which need to be solved by approximation. Composite projections use different projections depending on some threshold value. Yet other projections, such as the Robinson projection, define a table of predefined values, between which interpolation is used etc.. In each of these cases, we expect to find the operations corresponding to the mathematical structure of the projection in the transformation graph extracted by the presented method.

For verifying the method, we have implemented the patterns of several well-known cartographic projections based on the literature and have used it on the transformation graphs extracted from a variety of sample programs. To ensure a diversity of implementations, we have evaluated programs using different and independent JavaScript implementations of projections, including the open source libraries D3.js, proj4js, Leaflet, OpenLayers, and informal implementations of example programs found online. For these case studies, we could successfully identify many projections based on identifying patterns in the transformation graph in a fully automated, unsupervised manner.

In the future, the proposed method may be further developed for many innovative application scenarios, such as building a “cartographic search engine” or constructing novel tools for semi-automatic cartographic analysis and review.