CHAPTER V

HIERARCHICAL RESOLUTION

5.1 Overview

Our continuous cartogram method begins with a progressive hierarchical sampling, or simplification, of the resolution of the map regions. This is motivated primarily in the interest of making rapid overall adjustments to the map during the initial stages of the cartogram and then progressively moving toward more detailed (and time consuming) refinements. The map is initially sampled at a user-specified level of coarseness. Upon completion of the area and shape adjustments, the map is resampled at half the current coarseness. This hierarchical resampling continues until the user-specified number of simplifications has occurred.

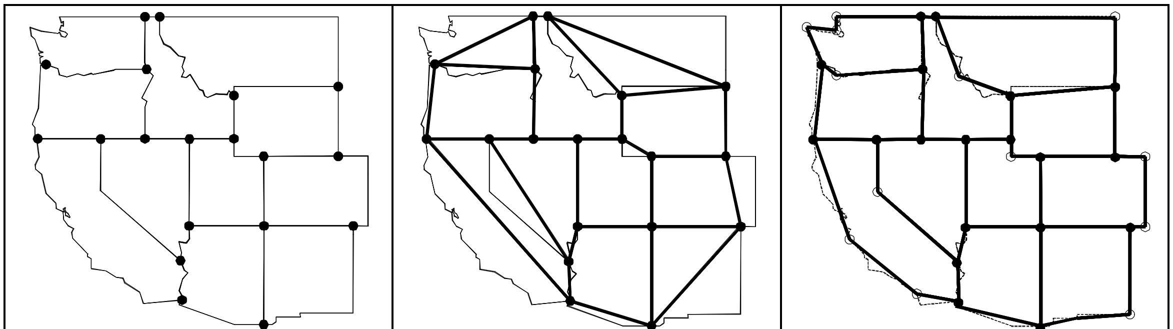
```
SimplifyMap (Regions, coarseness);

    IdentifyKeyPoints (Regions);
    for r = 1 to NumRegions
        allowedOffset := ComputeMaxOffsetDistance (Region[r], coarseness);
        { Simplify between each pair of Key Points }
        for p = 1 to Region[r].numKeyPoints
            keyPt := Region[r].KeyPoint[];
            if FoundSimplifiedEdge (Regions, keyPt[p], keyPt[p+1]) then
                CopySimplifiedEdge (Regions, keyPt[p], keyPt[p+1]);
            else { Construct minimum # of simplified lines btwn 2 key pts }
                AllocateNewSimplifiedKeyEdge (Region[r]);
                startPt := keyPt[p];
                endPt := keyPt[p+1];
                repeat
                    tempLine := MakeLine (startPt, endPt);
                    distance := FindMaxPointOffset (tempLine);
                    if distance > allowedOffset then
                        endPt := endPt – 1;  { Step back one vertex }
                    else
                        StoreSimplifiedLine (startPt, endPt);
                        StoreReconstructionInfo (startPt, endPt);
                        startPt := endPt;
                        endPt := keyPt[p+1];
                until startPt = keyPt[p+1];
```

Figure 5.1: Pseudocode of the simplification process in hierarchical resolution.

5.2 Identification of Key Points

The first step in the simplification pseudocode given in Figure 5.1 is the identification of certain shared vertices that would cause a break in the map topology, or connectivity, if they are simplified away. These "key points" can be defined as: 1) any interior point that is shared by three or more regions or 2) any point on the map perimeter that is shared by two or more regions. The key points of a western United States map are highlighted in Figure 5.2a.



a) Identification of "key" shared points in the western U.S.

b) Direct simplification between key points.

c) Simplification at a resolution relative to region sizes.

Figure 5.2: Examples of simplification during the hierarchical resolution process.

Conceptually, each state could be simplified as a set of edges connecting the key points, as demonstrated in Figure 5.2b. However, this is entirely impractical since the gross error in representing state sizes would nullify any benefit of the simplification. It is more desirable, as shown in Figure 5.2c, to simplify between key points at a resolution relative to the size of the region.

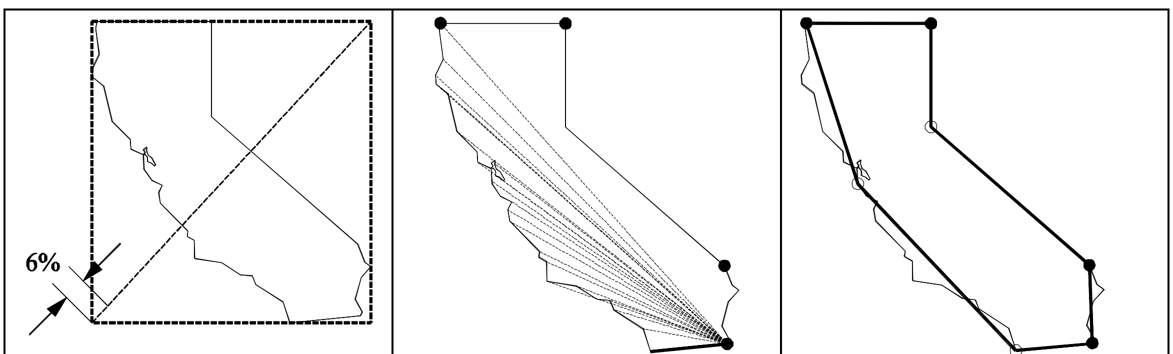
5.3 Simplification Between Key Points

A minimum number of simplified edges are constructed between two key points such that the distance between any vertex and its simplified edge is within an allowed offset distance. This distance is computed as a user-specified percentage of the length of the region's bounding box diagonal. In this manner the simplification is not based upon a global resolution but, instead, a collection of resolutions custom-scaled to each region.

This provides for a balanced simplification of region details that is independent of region size.

Prior to creating a simplified edge between two key points we check all other regions to see if this edge has already been simplified, as shown in the pseudocode. This is done not only for efficiency but out of necessity, since adjacent regions of varying size can produce vastly different results due to their differing offset distances and vertex traversal order. If a previous simplification between the two key points is found, it is stored and the focus is shifted to the next pair of key points. However, if a previous simplification is not found then the algorithm sets out to create the least number of simplified lines such that each vertex between the two key points is within the allowed offset distance from its simplified line.

The first line that is tested is the one directly connecting the two key points. If the minimum distance from the simplified line to any of the vertices between the two key points exceeds the allowed offset distance, the line endpoint is brought back one vertex and tried again. This continues until either an acceptable line is found or the edge is reduced down to two adjacent vertices.



a) Calculation of a 6% allowed maximum offset.

b) Process of testing proposed simplified edges.

c) Final simplified state of California.

Figure 5.3: Example process of simplifying the state of California.

Consider the California example depicted in Figure 5.3a, where we set the allowed maximum offset as 6% of the diagonal length. The first pair of key points bordered the coastline and a simplified line connecting them was tested first. Since at least one vertex was beyond the allowed distance from the line, a simplified line connecting the next point down the coastline was tested. This continued down the entire coast, as shown in Figure 5.3b, until the edge merely consisted of adjacent vertices spanning the bottom of the coast. Two more edges were required in order to complete the coastline simplification between this first pair of key points, as shown in Figure 5.3c. This “new” California enables rapid overall adjustments on account of its reduction from forty-seven edges down to *seven*!

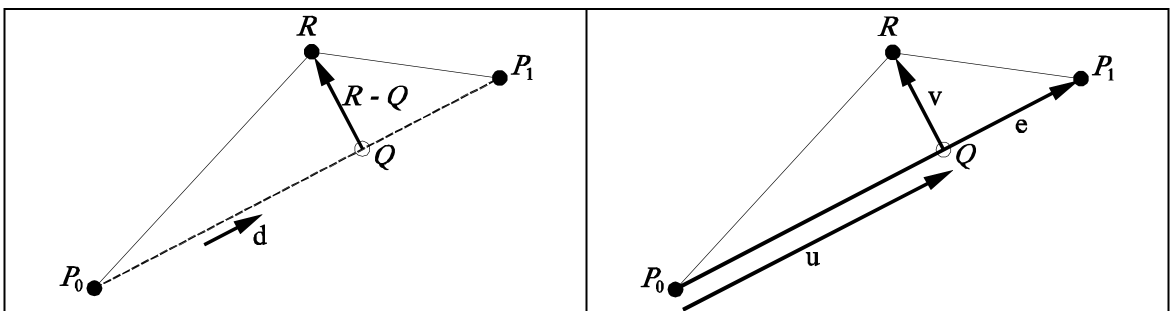
The minimum distance will always be along a path from the point to the line that is perpendicular to the line [8]. In the example depicted in Figure 5.4a, the minimum distance is at a point Q on the proposed line. The location of point Q between the line endpoints, P_0 and P_1, is

$$Q = P_0 + \left(\frac{(R - P_0) \cdot \mathbf{d}}{\mathbf{d} \cdot \mathbf{d}} \right) \mathbf{d}, \tag{5.1}$$

where **d** is a direction vector along the line. The minimum distance then becomes

$$\| R - Q \|, \tag{5.2}$$

the length of a vector from Q to R.



a) Finding the minimum distance from a vertex to the proposed simplified line.

b) Calculation of the perpendicular and parallel offset vectors.

Figure 5.4: Calculation of the offset distance and vectors.

5.4 Calculation of Reconstruction Information

Each simplified edge retains offset information to allow reconstruction (i.e. “unsimplification”) of its vertices regardless of how much the simplified edge has been scaled or rotated. As shown in Figure 5.4b, a parallel vector **u** and a perpendicular vector **v** are calculated for every vertex with respect to the origin of the simplified edge **e**. These vectors are used to compute the proportional offsets from the origin of the simplified line. The critical point Q is determined from equation 5.1.

The parallel vector **u** is simply the vector $Q - P_0$. Thus, the proportional offset value is

$$U = \|\mathbf{u}\| / \|\mathbf{e}\| . \quad (5.3)$$

It is important to note, however, that a vertex could lie *outside* the parallel extents of the line. In the example in Figure 5.4b, our vertex could have been situated to the left of P_0, incorrectly producing a positive offset proportion that, when reconstructed, would place the point *within* the extents of the line. This is easily corrected by negating U when the angle between **u** and **v** exceeds 90°. Similarly, the vertex could have been situated to the right of P_1, resulting in an offset value greater than 1 which, when reconstructed, places the vertex correctly beyond point P_1.

The perpendicular vector **v** is simply the vector $R - Q$. Thus, the proportional offset value is

$$V = \|\mathbf{v}\| / \|\mathbf{e}\| . \quad (5.4)$$

Note that this value does not indicate whether the vertex is to the left or right of the line. This differentiation can be made by negating V when the cross product $\mathbf{u} \times \mathbf{v}$ results in a negative direction.

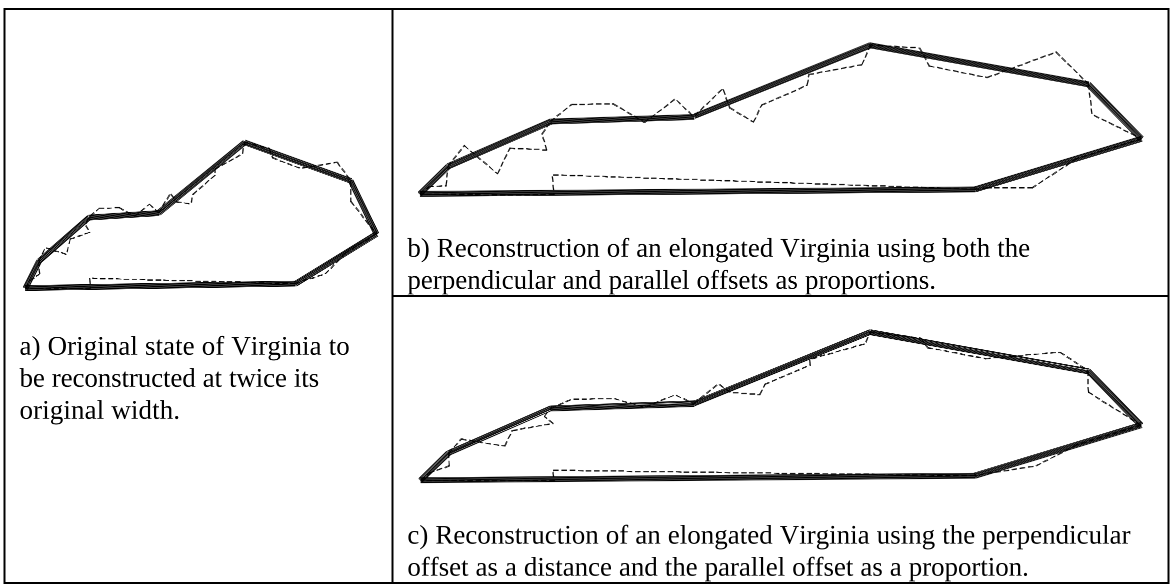
5.5 Reconstruction of Map Vertices

When no further improvements can be made to the cartogram at the current resolution, the entire map must be reconstructed before it can be resampled at a finer level of detail. We found that simply reconstructing the points in the exact parallel and perpendicular proportional offsets, however, is not desirable. What is the expected result, for example, when the simplified Virginia in Figure 5.5a is stretched to twice its width? By reconstructing in proportion to the elongated lengths of the simplified edges, we have greatly amplified the minor nooks and crannies of Virginia. These amplifications not only inhibit region recognition, as shown in Figure 5.5b, but can easily cause a self-intersection of the region.

A much better result is obtained for lengthened edges by reconstructing the perpendicular offset not as a proportion but instead as a measured distance, as demonstrated in Figure 5.5c. In situations where the simplified edge shortens, it is desirable to keep the perpendicular offset as a proportion so as not to amplify the nooks and crannies along this smaller line. Putting this together, the equation for the perpendicular offset vector with respect to the final edge $\mathbf{e}'$ becomes

$$\mathbf{v}' = \begin{cases} \|\mathbf{v}\|\mathbf{d}', & \|\mathbf{e}'\| > \|\mathbf{e}\| \\ V\|\mathbf{e}'\|\mathbf{d}', & \|\mathbf{e}'\| \le \|\mathbf{e}\|, \end{cases} \tag{5.5}$$

where $\mathbf{d}'$ is a unit vector pointing perpendicularly to the left of $\mathbf{e}'$.



a) Original state of Virginia to be reconstructed at twice its original width.

b) Reconstruction of an elongated Virginia using both the perpendicular and parallel offsets as proportions.

c) Reconstruction of an elongated Virginia using the perpendicular offset as a distance and the parallel offset as a proportion.

Figure 5.5: Example reconstructions of Virginia.

Since we desire to keep the parallel positioning of the vertices along the simplified line, the parallel offset vector is left as a proportion of the final edge $\mathbf{e}'$:

$$\mathbf{u}' = U\,\mathbf{e}'. \tag{5.6}$$

Using the offset vectors $\mathbf{u}'$ and $\mathbf{v}'$, each vertex is reconstructed from the origin of its simplified edge. Since reconstruction of the map to full resolution can easily produce topological inconsistencies such as self-intersected regions, intersection detection and removal is performed as detailed in Chapter VII until all intersections are removed. Then, if the user-specified number of simplifications has not yet been performed, the map is re-simplified at half the current level of coarseness. Otherwise, the resulting full resolution cartogram is stored as a final product.