Distortion-constrained Compression of Vector Maps

Alexander Kolesnikov
Department of Computer Science
University of Joensuu
P.O. Box, 80101
Joensuu, FINLAND
Phone: +358 46 810 4853
koles@cs.joensuu.fi

Alexander Akimov
Department of Computer Science
University of Joensuu
P.O. Box, 80101
Joensuu, FINLAND
Phone: +358 13 251 7903
akimov@cs.joensuu.fi

ABSTRACT
An algorithm for lossy compression of vector maps for given error tolerance was developed. The algorithm is based on optimal polygonal approximation and dynamic quantization of vector data. A near optimal distortion-constrained quantizer with step defined by the tolerance level was constructed. The proposed algorithm performed well compared to other approaches.

Categories and Subject Descriptors
G.1.2 [Numerical Analysis] Approximation: Approximation of surfaces and contours

General Terms
Algorithms

Keywords
vector map, compression, polygonal approximation, quantization, simplification.

1. INTRODUCTION
The compression of vector data is necessary for the efficient storage and transmission of vector maps to end-user devices, such as desktops, notebooks or PDA units. To reduce storage space and transmission time, vector maps can be compressed. Some loss in the accuracy of the representation of vector data is acceptable if the distortion does not exceed a certain level $\varepsilon$ (see Figure 1).

Data reduction in vector maps can be achieved in two ways: 1) through polygonal approximation (simplification) [2-7,9,10,16] and 2) through the quantization of vector data [1,11,14,15]. After having been processing, vector data can be compressed with an entropy encoder, based on the data’s statistics. With polygonal approximation, the number of vertices in a vector map is reduced to a minimal value, keeping the approximation error below a given threshold $\varepsilon$. There are a number of heuristic algorithms for polygonal approximation [5,10,16]. The optimal solution to the problem of finding the minimum number of segments ($\text{min-# problem}$) [4,6,7] and the minimum bitrate ($\text{min-rate problem}$) [12,13] can be found by determining the shortest path in a weighted graph (see Figure 2). In [12,13], the compression of digitized contours was discussed. The amount of data in a vector map can also be reduced through the quantization of coordinates. The reduction of data through vector quantization of map coordinates with a fixed-size codebook was discussed in [1,14,15]. In [11] was represented the vector data reduction by uniform quantization of map coordinates.

For better compression performance, real-valued coordinates of vector maps should be quantized and polygonal approximation should be done, while keeping the total distortion below the threshold. The quantization of vector data can be performed before, after, or during polygonal approximation.

In [9], an algorithm for $\text{min-rate}$ polygonal approximation with embedded vector quantization was introduced. That paper also described how the algorithm could be used for vector map compression. In this paper we expand on the ideas introduced in [9]; namely, we present a practical algorithm for distortion-constrained minimal-entropy compression of vector maps including design of a distortion-constrained minimum-entropy quantizer for the vector data.

Figure 1. Fragment of input vector map # 1, file size is 534 Kbytes (left), and result of compression for $\varepsilon=2.5$ m, file size is 18.5 Kbytes (right).

2. DISTORTION-CONSTRAINED POLYGONAL APPROXIMATION
2.1 Problem formulation
The problem of optimal distortion-constrained polygonal approximation can be formulated as follows [4,6,7,12,13]: Given an open $N$-vertex polygonal curve $P = \{p_1, \ldots, p_N\}$, approximate the curve using another polygonal curve $Q$ with the minimum number of linear segments $M$ ($\text{min-# problem}$) or minimal bitrate $R$ ($\text{min-rate problem}$) so that the approximation error with error measure $L_\infty$ does not exceed a given tolerance level $\varepsilon$. To solve this problem, first a feasibility graph $G = G(P, \varepsilon)$ is constructed on...
the vertices of the input curve for a given error tolerance \( \varepsilon \) (see Figure 2). An edge \( v(i,j) \) of graph \( G \) corresponds to the approximation of a curve segment \( \{p(i), ..., p(j)\} \) with a line segment passing the terminal points \( p(i) \) and \( p(j) \). Nodes \( n_i \) and \( n_j \) of \( G \) are connected with edge \( v(i,j) \) if the corresponding polygonal approximation error does not exceed the given threshold \( \varepsilon \).

The weight \( w(i,j) \) assigned to the edge \( v(i,j) \) depends on the problem type. For the min-\# problem, the weight is equal to 1, \( w(i,j)=1 \); for the min-rate problem, the weight is defined by the number of bits needed for the encoding of the line segment with an entropy encoder: \( w(i,j)=\Delta r(i,j) \). The optimal solution for the problem in question can be found: it is the shortest path in the weighted graph \( G \) (see Figure 2).

The complexity of the algorithm for min-\# and min-rate problems depends on complexity of algorithm for the feasibility graph construction, which is \( O(N^2) \) according to [4], and on the complexity of the algorithm for the shortest path construction in the directed acyclic graph, which is \( O(N^2) \) too. The complexity of the bitrate calculation for a single edge is \( O(1) \). In total, the time complexity is \( O(N^2) \).

2.2 The prequantization approach

The quantization of vector data can be performed before, after, or during polygonal approximation. Here we discuss both pre- and postquantization approaches. With the former approach, vertex coordinates are first quantized with step \( q \), then an optimal algorithm is applied for min-\# or min-rate polygonal approximation of the quantized vertices (see Figure 3).

To satisfy a given constraint on the total distortion, the maximum distortion, \( \delta \), caused by the prequantization of two-dimensional coordinates should be less than the given error tolerance \( \varepsilon \):

\[
\delta = \sqrt{\frac{q^2}{2}} + \sqrt{\frac{q^2}{2}} \leq \varepsilon.
\]

Equation 1 constrains step \( q \) as follows: \( q \leq \sqrt{\varepsilon} \). Integer values of quantized coordinates can be compressed with an entropy encoder. The upper limit of the step, \( q = \varepsilon \sqrt{2} \), is used because with a larger quantization step less bits are needed to encode the results of quantization than with a smaller quantization step.

2.3 The postquantization approach

With the postquantization approach, the min-\# polygonal approximation is first constructed, then the coordinates of the approximating curve are quantized with step \( q \) (see Figure 3). This approach has an essential drawback: the sum of distortions caused by quantization and polygonal approximation exceed the tolerance level \( \varepsilon \) \((\delta > \varepsilon)\), even when the polygonal approximation error was below the threshold \( \varepsilon \).

3. POLYGONAL APPROXIMATION with DYNAMIC QUANTIZATION

3.1 In this section we discuss polygonal approximation with embedded quantization [9]. For the sake of simplicity, we refer to this approach as dynamic quantization because the quantization is performed dynamically during the search for the optimal polygonal approximation.

3.1 Polygonal approximation error

Two-dimensional relative coordinates, \( \Delta p_{ij} \), for the two vertices, \( p(i) \) and \( p(j) \), are defined as difference between the vertices: \( \Delta p_{ij} = p(j) - p(i) \). Closed-loop quantization of the relative coordinates is used to avoid accumulation of quantization error. The relative coordinates \( \Delta p_{ij} \) for the pair of vertices are defined as difference between the vertex \( p(j) \) and the restored value \( p_{r}(i) \) of the vertex \( p(i) \): \( \Delta p_{ij} = p(j) - p_{r}(i) \). The restored point \( p_{r}(i) \) is defined by the restored vertex \( p_{r}(i) \) and by the quantized value of the relative coordinates: \( p_{r}(i) = p(i) + Z \Delta p_{ij} \) (see Figure 4). If the coordinates of the starting points of the curves are quantized with a two-dimensional uniform quantizer, then the restored points of the approximating curve take values on the uniform grid.

Quantization of the relative coordinates induces displacement of the restored approximating nodes \( p_{r}(j) \) relative to the original vertices \( p(j) \) and affects the error of polygonal approximation. The distance from vertex, \( p(k) = (x_k, y_k) \in P \), to the approximating line \( L(p_{r}(i), p_{r}(j)) \) passing the restored vertices, \( p_{r}(i) \) and \( p_{r}(j) \), is given by the following expression:

\[
d(p(k); (p_{r}(i), p_{r}(j))) = \frac{|ax_k + by_k + c|}{\sqrt{a^2 + b^2}},
\]

here the coefficients \( a, b, \) and \( c \) of the line can be computed from the restored values \( p_{r}(i) \) and \( p_{r}(j) \) as the end points of the line segment.

The approximation error \( e_{\varepsilon}(p_{r}(i), p_{r}(j)) \) for a curve segment \( \{p(i), ..., p(j)\} \) with measure \( L_{\omega} \) is defined as the maximal
distance from vertices $p(k)$ of the curve segment to the approximating line, $L(p(i), p(j))$ (see Figure 4):

$$e_{\text{approx}}(p(i), p(j)) = \max_{i \leq k \leq j} \{ d(p(k); (p(i), p(j))) \}. \quad (2)$$

Figure 4. A curve segment (white circles) is approximated by a line segment.

The total distortion $E_{\text{tot}}(P)$ for the input curve $P$ approximated by curve $Q$ is defined as the maximum of deviations $e_{\text{approx}}(i, j)$ for all segments:

$$E_{\text{tot}}(P) = \max_{i \leq k \leq j} \{ e_{\text{approx}}(p(i), p(j)) \} \quad (3)$$

where $p(i)$ and $p(j)$ are restored values for the terminal points of the curve segment $\{p(i), ..., p(j)\}$, and $M$ is the number of line segments. The complexity of the algorithm for the calculation of maximum deviation $E_{\text{tot}}(P)$ is $O(N^2)$.

### 3.2 Optimal polygonal approximation

The problem of optimal distortion-constrained minimum-entropy polygonal approximation with dynamic quantization of relative coordinates can be stated as follows: approximate the polygonal curve $P$ by the polygonal curve $Q$ so that total bit-rate $R(P, \varepsilon)$ is minimized and approximation error does not exceed the given threshold:

$$R(P, \varepsilon) = \min_{i=1}^{N} \sum_{m=1}^{M} \Delta \varepsilon(p(i), p(j)) \quad (4)$$

subject to: $E_{\text{approx}}(P) \leq \varepsilon$.

The solution to the optimization problem is shortest path in the weighted graph $G$ constructed on the vertices of $P$. The edge $v(i, j)$ connecting the two nodes $m(i)$ and $n(j)$ of $G$ corresponds to the approximation of curve segment $\{p(i), ..., p(j)\}$ by the line segment $L(p(i), p(j))$. The weight $w(i, j)$ of the edge $v(i, j)$ in $G$ is defined by the bitrate for encoding the line segment as follows:

$$w(i, j) = \begin{cases} \Delta \varepsilon(p(i), p(j)), & \text{if } e_{\text{approx}}(p(i), p(j)) \leq \varepsilon; \\ \infty, & \text{otherwise.} \end{cases}$$

The bitrate $\Delta \varepsilon(p(i), p(j))$ for encoding the two-dimensional relative coordinates $p(j)-p(i)=\Delta \varepsilon(\Delta x, \Delta y)$ is estimated from the corresponding probability distributions $f_1(\Delta x)$ and $f_2(\Delta y)$:

$$\Delta \varepsilon(p(i), p(j)) = -f_1(\Delta x) \cdot \log_2 f_1(\Delta x) - f_2(\Delta y) \cdot \log_2 f_2(\Delta y),$$

here we assume that relative coordinates $\Delta x$ and $\Delta y$ are statistically independent.

Let’s define cost function $C(n)$ as minimal bitrate for compression of subcurve $\{p(1), ..., p(n)\}$ to find the shortest path in the weighted acyclic graph. Then the minimal bitrate for all $n=1, ..., N$ can be found with the following recursive equation:

$$C(n) = \min_{i, j < n} \{ C(j) + \Delta \varepsilon(p(j), p(n)) \}; \quad C(1) = 0.$$ 

The feasibility graph $G$ cannot be constructed beforehand because quantization of the relative coordinates has to be performed during the search for the shortest path. The complexity of the algorithm is defined by the complexity of the shortest path construction in a directed acyclic graph, which is $O(N^2)$, and by linear complexity for error calculation. In total, the complexity is $O(N^3)$.

Figure 5. Example of min-# polygonal approximation with dynamic quantization.

For the min-# problem, the weight of the edge is equal to one, $w(i, j)=1$, if the distortion for the edge does not exceed tolerance level $\varepsilon$, otherwise the edge is not feasible: $w(i, j)=\infty$. On Figure 5 is shown example of min-# polygonal approximation with dynamic quantization for the test shape.

The solution for the alternative problem of entropy-constrained minimal-error compression of vector maps can be found using the introduced algorithm for distortion-constrained vector map compression. A binary search is conducted in $O(\log(N))$ steps to find such a distortion $\varepsilon$ that the bitrate $R(P, \varepsilon)$ is below a given bitrate threshold $R_0$: $R(P, \varepsilon) < R_0$.

### 3.3 Distortion-constrained quantization

For min-rate polygonal approximation with dynamic quantization [9] a vector quantizer that satisfies two conditions is needed: 1) the bitrate is minimal, and 2) the maximum quantization error is constrained. Assuming that in vector maps the relative coordinates $\Delta x$ and $\Delta y$ are statistically independent, two scalar quantizers for the coordinates have to be constructed individually.

The problem of distortion-constrained minimal-rate scalar quantization is similar to the problem of distortion-constrained minimal-rate polygonal approximation (more precisely, polygonal approximation of convex curves). The optimal solution to the scalar quantization problem can be found as the shortest path in the weighted feasibility graph $G_q$ with the one-dimensional relative coordinates as nodes. The codebook should be stored in an output file, or be reconstructed on-the-fly using current statistics. With the former approach, the compression performance is poor, especially when there is little distortion. The latter approach is time-consuming.

There is a more simple solution to the problem. The uniform quantizer with the step $q=\sqrt{2} \varepsilon$ satisfies the cell constraints, by definition. However, the superiority of the optimal distortion-constrained minimum-entropy quantizer over the uniform quantizer is relatively small [17]. Indeed, because of the constraint on the maximal size of cell, the only way to reduce bitrate is to decrease the size of some quantization cells. On other
hand, the cells cannot be decreased too much because further decreasing of the cell size leads to larger entropy. Moreover, if the costs for codebook storage are taken into consideration, uniform quantization seems to be a good and practical solution for the problem at hand.

3.4 Entropy encoding of quantized coordinates
For entropy encoding of quantized relative coordinates probability distribution of the coordinates should be known. We take the following approach: start processing the data using a uniform distribution of the relative coordinates and update the probability distribution using set of relative coordinates of the processed curves. The frequency histogram is first used in min-rate polygonal approximation for bitrate evaluation and then for the compression of quantized relative coordinates.

For data decoding, we follow the same scheme: we start processing the data using a uniform distribution of the relative coordinates and update the probability distribution using set of relative coordinates of the restored curves.

4. EXPERIMENTS and DISCUSSIONS
A test set of vector data; the vector maps Elevation lines and Europe; is presented in Figure 6. The data are given in the ESRI/ArcShape file format with an accuracy of 16 bytes per vertex.

Figure 6. Test set of vector maps: 1) Elevation lines, \(N=20184\) vertices, file size 534 Kbytes; 2) Europe, \(N=169673\) vertices, file size 2671 Kbytes.

We tested the prequantization approach with the heuristic Douglas-Peucker algorithm \[5\] and with the optimal algorithms for the \(\text{min-}\#\) and \(\text{min-rate}\) problems \[4,12,13\]: PreQ_Douglas-Peucker, PreQ_min-\# and PreQ_min-rate algorithms. The dynamic quantization method was tested for the \(\text{min-}\#\) and \(\text{min-rate}\) problems; the DynQ_min-\# and DynQ_min-rate algorithms were used, respectively, for the \(\text{min-}\#\) and \(\text{min-rate}\) problems.

The results of the experiments are presented in Figures 7 and 8. The output file included the beginnings of curves and the relative coordinates of approximation nodes (line segments). The data reduction was performed by reducing the number of points in the curves and through the quantization of coordinates. To evaluate the contribution of quantization in the reduction of the output file size, the bitrate per approximating line segment was also calculated. The processing time was about one second for the Elevation lines map and about one minute for the Europe map.

For small values of the tolerance level \(\varepsilon\), the reduction in the number of vertices caused by polygonal approximation was relatively small. The main contribution to the data reduction comes from the quantization of the relative coordinates; for test maps the bitrate decreased from an original 16 bytes per vertex to 1.5 bytes in the compressed file.

Figure 7. Rate-distortion curves for the vector map #1: the output file size (top) and bitrate per line segment (bottom) as function of distortion \(\varepsilon\).

Figure 8. Rate-distortion curves for the vector map #2: the output file size (top) and bitrate per line segment (bottom) as function of distortion \(\varepsilon\).
When distortion $\varepsilon$ was increased, the elimination of vertices caused a further reduction in data. For example, for the map #1 the number of vertices was decreased in 10.30 times meanwhile the bitrate per line segment decreased only from 10 bits to 7.5 bits.

An analysis of the total bitrate shows that the optimal algorithm with dynamic quantization provides better compression performance than those with prequantization, except for the case of near-lossless compression for very small values of $\varepsilon$.

As it was expected, the optimal min-# polygonal approximation leads to better results than the heuristic Douglas-Peucker algorithm. The situation with individual bitrate for approximating line segments is exactly inverse to those for the total bitrate (file size). Douglas-Peucker algorithm gives the smallest bitrate per line segment because of the larger number of total bitrate (file size). Douglas-Peucker algorithm gives the approximating line segments is exactly inverse to those for the approximating line segments.

Peucker algorithm. The situation with individual bitrate for approximation leads to better results than the heuristic Douglas-Peucker algorithm. Some possible explanations for this phenomenon follow:

1. The total bitrate depends more on the number of the line segments than on the bitrate for individual line segments.
2. The statistic we used to calculate the cost function for min-rate polygonal approximation was not the same as the one we used later for the quantized data compression.

So, for the tested vector maps the total bitrate depends more on the structure of the feasibility graph $G$ than on the concrete weights of the edges. As was shown before, the bitrate per line does not change very much for different values of distortion $\varepsilon$. From a practical point of view, this is an important feature. It means that efficient compression of vector maps can be obtained through the simpler min-# polygonal approximation.

5. CONCLUSIONS
To solve the problem of lossy compression of vector data for given error tolerance an optimal algorithm was developed. The algorithm is based on optimal polygonal approximation with dynamic quantization of vector data. An uniform near-optimal distortion-constrained quantizer with step $q$ defined by the tolerance level $\varepsilon$ was constructed. The proposed algorithm performed well compared to other approaches on the tested maps.

In terms of practical applications, the problem of topological consistency of simplified maps should be taken into account. A topic for the further research is the compression of vector maps with progressive transmission [2,3] combining the approach presented here and the optimal algorithm for multi-resolution polygonal approximation we developed earlier [8].

6. REFERENCES